RAMSIM: A fast computer model for mean wind flow over hills

Corbett, Jean-Francois

Publication date:
2007

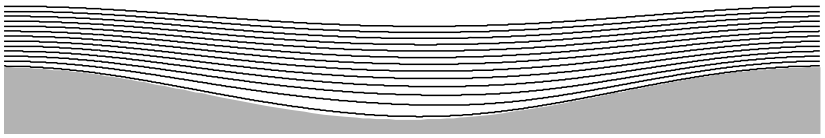
Document Version
Publisher's PDF, also known as Version of record

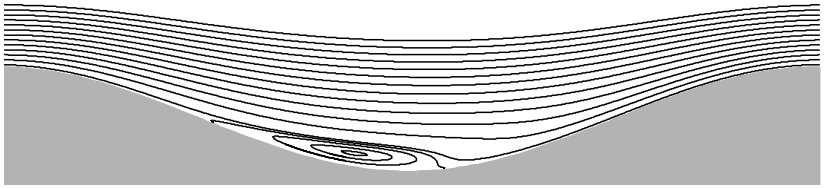
Risø-PhD-thesis

RAMSIM
A fast computer model for mean wind flow over hills

Jean-François Corbett

Risø-PhD-17(EN)





Risø National Laboratory
Technical University of Denmark
Roskilde, Denmark
June 2007

Author: Jean-François Corbett
Title: RAMSIM: A fast computer model for mean wind flow over hills
Department: Wind Energy Department

Risø-PhD-17(EN)
June 2007

This dissertation was submitted to the University of Copenhagen in December 2006 in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Physics, and was successfully defended on March 9th, 2007. Minor corrections and changes were brought prior to release as the present Risø report. The original dissertation was entitled, "A novel approach for the fast modeling of mean wind flow over complex orography."

Abstract:
The Risø Atmospheric Mixed Spectral-Integration Model (RAMSIM) is a micro-scale, linear flow model developed to quickly calculate the mean wind flow field over orography. It was designed to bridge the gap between WAsP and similar models that are fast but insufficiently accurate over steep slopes, and non-linear CFD models that are accurate but too computationally expensive for routine use on a PC.

RAMSIM is governed by the RANS and E-ε turbulence closure equations, expressed in non-Cartesian coordinates. A terrain-following coordinate system is created from a simple analytical expression. The equations are linearized by a perturbation expansion about the flat-terrain case. The first-order equations, representing the spatial correction due to the presence of orography, are Fourier-transformed analytically in the two horizontal dimensions. The pressure and horizontal velocity components are eliminated, resulting in a set of four ordinary differential equations (ODEs). RAMSIM is currently implemented and tested in two-dimensional space; a 3D version has been formulated but not yet implemented.

In the 2D case, there are only three ODEs, depending on only two non-dimensional parameters. This is exploited by solving the ODEs by Runge-Kutta integration for all useful combinations of these parameters, and storing the results in look-up tables (LUT). The flow field over any given orography is then quickly obtained by interpolating from the LUTs and scaling the value of the flow variables for each wavenumber component of the orography, and returning to real space by inverse Fourier transform.

RAMSIM was tested against measurements, as well as other authors' flow models, in four test cases: two laboratory flows over idealized terrain, and two field experiments. RAMSIM calculations generally agree with measurements over upward slopes and hilltops, but overestimate the speed very near the ground at hilltops. RAMSIM appears to have an edge over other linear models in lee-side wind speed calculations and in predicting the occurrence of a recirculation region (though underestimating its size by half). RAMSIM is able to predict asymmetric flow over symmetric hills, while WAsP is not. Non-linear CFD models are more accurate than RAMSIM, but orders of magnitude more expensive computationally.

ISBN 87-550-3488-8

Group's own reg. no.:
1191005-01

Cover :
Streamline plots of the flow field over sinusoidal hills, as calculated by RAMSIM, illustrating the onset of flow separation as the maximum slope is increased from 20% (top) to 40% (bottom).

Pages: 140
Tables: 4
References: 88

Information Service Department
Risø National Laboratory
Technical University of Denmark
P.O.Box 49
DK-4000 Roskilde
Denmark
Telephone +45 46774004
bibl@risoe.dk
Fax +45 46774013
www.risoe.dk

Acknowledgements

I am eternally grateful to Søren Ott, who allowed me to tap the unfathomable depths of his scientific knowledge, and provided invaluable guidance at the time when I needed it most. I also wish to thank my Risø supervisor Lars Landberg for giving me the opportunity to start and, in spite of everything, finish this PhD; and also Aksel Walløe Hansen, my official supervisor at Københavns Universitet. Many thanks to my friends and colleagues in the MET program, who were always available for scientific and technical help, and who gave me a reason to show up at work even through the difficult times. I would not be writing these lines had it not been for the tremendous, unfailing support of my family, as well as that of my friends who were in the same boat as I. I am grateful to my wife Liv for bearing with me through the rough seas, and giving meaning to this entire adventure.

Nobody said it was easy
No one ever said it would be this hard
I'm going back to the start
Ahhooooooooooooooooo
Ahhooooooooooooooooo

Coldplay – The Scientist

Contents

List of Figures

List of Tables

Nomenclature

Note: Variables used only locally are not exhaustively listed here.

Roman letters

$\mathbf{A}$	matrix of coefficients of the system of ODEs
$\mathbf{A}_\infty$	matrix of coefficients $\mathbf{A}$ in the limit $k\acute{z} \to \infty$
A	$= \mathsf{i}k_q\acute{U}_q^{(0)}$
B	$= \mathsf{i}\breve{k}_q\acute{U}_q^{(0)}$
$C_\mu, C_{\varepsilon 1}, C_{\varepsilon 2}$	constants of the $E-\varepsilon$ turbulence model
$\mathcal{D}$	Fourier-transformed first-order dissipation of TKE (shorthand for $\hat{\varepsilon}_1$)
E	Turbulent kinetic energy (TKE)
$\mathcal{E}$	Fourier-transformed first-order TKE (shorthand for $\hat{E}_1$)
f_c	Coriolis parameter
$\mathcal{F}\{\star\}$	horizontal Fourier-transform operator
g	gravitational acceleration
h	height above ground level
h_BL	depth of the boundary layer
h_t	terrain height
i	the imaginary number $\sqrt{-1}$
$\Im\{\phi\}$	imaginary part of complex number ϕ
J	$\equiv \det \frac{\partial x_i}{\partial \acute{x}_j}$, the Jacobian of the coordinate transformation
k_i	wavenumber in coordinate $\acute{x}_i$, $i = 1, 2$
k	wavenumber: in the 2D case, we drop the subscript: $k = k_1$
$\mathbf{k}$	horizontal wavenumber vector: $\mathbf{k} = (k_1, k_2)$
$\breve{k}_i$	perpendicular wavenumber: $(\breve{k}_1, \breve{k}_2) \equiv (-k_2, k_1)$
L_buf	buffering length appended to each end of the domain
l_mix	mixing length
l_max	maximum mixing length
L_σ	length scale used to non-dimensionalize the ODEs
$\mathbf{M}$	coefficient matrix of the lower boundary conditions on the first-order quantities
N_v	number of dependent variables in the system of ODEs
n_1, n_2	number of boundary conditions specified at the lower and upper boundaries, respectively
P	pressure
p	hydrostatic pressure (referred to simply as "pressure" in most of this document)
$\mathbf{Q}$	coefficient matrix of the upper boundary conditions on the first-order quantities
$\Re\{\phi\}$	real part of complex number ϕ
s	expansion parameter
S_i	source terms of the i^th component of the momentum equation
S_E, S_ε	source terms of the of TKE and dissipation transport equations, respectively
$\mathbf{S}$	matrix of source terms in the system of ODEs
ΔS	fractional speedup
T_σ	time scale used to non-dimensionalize the ODEs
u_i	ith component of the velocity: $(u_1, u_2, u_3) = (u, v, w)$
$\acute{U}_i$	ith component of the velocity in transformed coordinates (normalized by the jacobian)
$\mathcal{W}$	Fourier-transformed first-order vertical velocity (shorthand for $\hat{\acute{U}}_3^{(1)}$)
x_i	ith Cartesian coordinate: $(x_1, x_2, x_3) = (x, y, z)$
$\acute{x}_i$	ith transformed coordinate: $(\acute{x}_1, \acute{x}_2, \acute{x}_3) = (\acute{x}, \acute{y}, \acute{z})$
$\acute{\mathbf{x}}$	transformed horizontal position vector: $\acute{\mathbf{x}} = (\acute{x}_1, \acute{x}_2)$
$\mathbf{X}$	matrix of dependent variables in the system of ODEs

$\acute{z}$	"vertical" coordinate in the terrain-following transformed coordinate system
z_0	surface roughness length
$\acute{z}_{\mathrm{fit}}$	location of the fitting point (solving ODEs by shooting to a fitting point)

Greek letters

α_i^j	$\equiv \frac{\partial \acute{x}_j}{\partial x_i} = \left(\frac{\partial x_i}{\partial \acute{x}_j}\right)^{-1}$
β_j^i	$\equiv \frac{\partial x_i}{\partial \acute{x}_j}$
γ	angle between the upper-boundary forcing stress and the wavenumber vector
δ_{ij}	Kronecker delta
ε	dissipation of turbulent kinetic energy
ϵ_{ijk}	Levi-Civita permutation symbol
θ_{f}	azimuthal angle of the upper-boundary forcing stress
θ_{m}	azimuthal wind direction at a reference point in the domain
κ	von Karman constant
λ_i	coordinate transformation in dimension i: $x_i = \acute{x}_i + \lambda_i$
Λ_i	coordinate transformation in the Fourier domain: $\Lambda_i = \mathcal{F}\{\lambda_i\}$
ν	kinematic viscosity
ν_{m}	kinematic molecular viscosity
ν_{t}	kinematic eddy viscosity
Π	production of turbulent kinetic energy
ρ	density of air
σ	standard deviation of the Gaussian smoothing filter applied to the terrain
$\sigma_E, \sigma_\varepsilon$	constants of the $E-\varepsilon$ turbulence model
τ_{ij}	$= \overline{u_i' u_j'}$ Reynolds stress tensor
ϕ	dummy variable
$\boldsymbol{\omega}$	vorticity vector
$\acute{\omega}_3$	vertical component of the vorticity in the transformed coordinates
Ω	Fourier-transformed first-order vertical vorticity (shorthand for $\hat{\acute{\omega}}_3^{(1)}$)

Modifiers

These subscripts, superscripts, etc., modify the meaning of quantity ϕ:

$\overline{\phi}, \phi'$	mean and fluctuating parts of ϕ (in the context of Reynolds averaging)
$\acute{\phi}$	in transformed coordinates
$\hat{\phi}$	Fourier-transformed horizontally (along $\acute{x}$ and $\acute{y}$)
ϕ'	vertical derivative $\partial\phi/\partial\acute{z}$
ϕ''	second vertical derivative $\partial^2\phi/\partial\acute{z}^2$
ϕ'''	third vertical derivative $\partial^3\phi/\partial\acute{z}^3$
ϕ''''	fourth vertical derivative $\partial^4\phi/\partial\acute{z}^4$
ϕ_0, ϕ_1	scalar: zero-order (basic flow), first-order perturbation
$\phi_i^{(0)}, \phi_i^{(1)}$	vector: zero-order (basic flow), first-order perturbation
$\phi^j_{i\,(0)}, \phi^j_{i\,(1)}$	metrics α and β: zero-order (basic flow), first-order perturbation
	For vectors and metrics, the order number is placed in brackets to avoid confusion with the other subscripts and superscripts.

Acronyms

ABL	atmospheric boundary layer
a.g.l.	above ground level
BC	boundary condition

FFT	Fast Fourier Transform
LHS	Left-hand side (of equation)
MSFD	mixed spectral finite-difference
MSI	mixed spectral integration
ODE	Ordinary differential equation
RAMSIM	Risø Atmospheric Mixed Spectral-Integration Model
PC	personal computer
RANS	Reynolds-averaged Navier-Stokes
RHS	Right-hand side (of equation)
TKE	turbulent kinetic energy
WA^{S}P	Wind Atlas Analysis and Application Program

Units

kWh	kilowatt-hour
MW	megawatt

1 Introduction

Over the past decade, the exploitation of wind as a source of energy has received increasing support as a clean and viable alternative to energy extraction from fossil and nuclear fuels. The motivation behind this push is, to be sure, in part a rational concern over the planet-wide consequences of the continued burning of fossil fuels, and over the safety issues of nuclear fuel and waste. Added to this, however, is the growing insecurity of supply and upward trend of the cost of fossil fuels, a trend that can only be expected to steepen, as the world's most easily accessible deposits become depleted and the fossil fuel industry moves on to sources that are more difficult and costly to exploit. In contrast, the cost of wind energy has been falling steadily, due to the tremendous technological progress that has marked this relatively young industry: in the last twenty years, the cost per kilowatt-hour (kWh) has fallen by more than half. Wind is therefore standing up as an increasingly advantageous energy source, and is bound to play an important role in the global energy industry of the twenty-first century.

Since the mid-1990's, wind power has enjoyed spectacular expansion, and continues to be a rapidly growing sector of energy production worldwide, with two-digit annual growth rates. At the end of 2005, the installed global wind power capacity stood at 59,000 megawatts (MW), 11,000 of which were installed in 2005. The mass deployment of wind turbines in the late 90's and early triple 0's occurred mainly in a few European countries: Denmark, which led the way, and has long held the world's highest proportion of national electricity production by wind, now at approximately 20%; Germany, which currently has the largest wind power deployment in the world, at almost 20,000 MW; and Spain, currently producing 8% of its electricity from wind. Recent years, however, have seen a surge of growth in Asia, with India in the lead; and in North America, where even the author's own country of birth, Canada, by all accounts a late starter, has increased its wind energy capacity by a staggering 54% in 2005 – though admittedly only to an modest total of 683 MW. (All figures from Global Wind Energy Council, 2005.)

Continued, effective growth into the coming decades will depend on a number of social and political factors, but also on technical development in many areas, including: wind turbine design and materials, incorporation of wind-generated power into the power grid, and, not least, wind power meteorology.

1.1 Wind power meteorology

The field of wind power meteorology has the goal of understanding all the atmospheric phenomena that may affect the production of energy using wind turbines. For a complete review of the field, the reader is directed to Petersen et al. (1998a) and (1998b). Wind power meteorology addresses three main challenges:

Forecasting of the wind phenomena susceptible to affect the production of wind power in the one- to ten-hour horizon. This is vital for the efficient coordination of wind, a fluctuating source of power, with the other power generation modalities on the grid, which often have significant start-up and shut-down lag times. Firing up a coal-fired power plant can take up to eight hours, while a nuclear power plant can take the better part of a day.

Regional assessment of the wind resource A regional assessment is intended to reflect the potential output of an large number of turbines deployed across this region. Increasingly, resource assessment has been accomplished with the help of numerical wind flow models. Ideally, the assessment produces a re-

source map with a high resolution, down to tens of kilometers, with information on the distribution of wind speed and direction, as well as diurnal and seasonal variability. This is achieved using large- and meso-scale climate models (domain size $\sim 10^4$ to 10^3 km).

Local assessment of the wind resource a.k.a. siting Local topographical features can have a significant impact not only on the output of a turbine deployed at a specific site, but also on its durability. A micro-scale wind flow model (domain size $\sim 10^1$ km), with local measurements as an input, can yield information on the local wind resource, as well as on turbulence conditions and the effect of rotor wakes.

Accurate resource prediction is essential in optimizing the placement of wind turbines, as well as in estimating the potential returns of a given wind farm project in relation to the investment required.

A current challenge for the wind energy industry is the prediction of the wind resource in complex terrain. There is interest in placing wind turbines in hilly and mountainous areas, due to the potentially high speed-up in certain areas, leading to a locally high wind resource. Currently missing, however, is a flow model that is both accurate in complex terrain, and simple and computationally cheap enough to be used on a routine basis on personal computers. The development of such a model is the subject of the present thesis.

1.2 Wind resource prediction

Importance of accuracy

Accuracy is paramount in wind resource prediction, since the power output of a wind turbine can vary sharply with wind speed. The wind power available for extraction by a turbine varies in proportion to the cube of the wind speed. This is easily shown by considering a small mass of air dm of density ρ passing through a given rotor area A_r in a short time interval dt (Figure 1). For simplicity, we

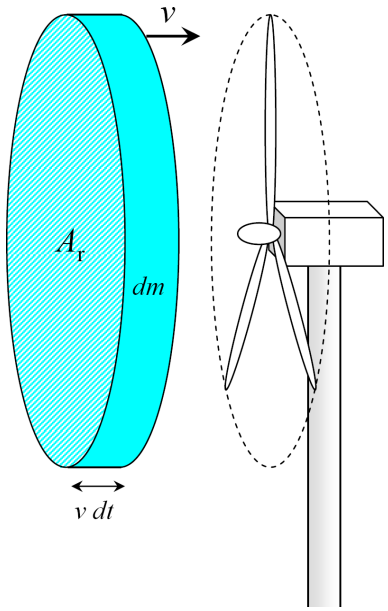


Figure 1: A mass of air dm passing through a rotor area A_r at speed v in a time interval dt.

assume that the air mass has a uniform speed v, though in reality it is not uniform, especially due to a vertical speed gradient caused by friction at the ground surface. The air mass carries a kinetic energy dE_k that is proportional to the square of its speed:

$$dE_k = \frac{1}{2}\, dm\, v^2$$

The size of the mass of air is in turn proportional to v:

$$dm = \rho A_\mathrm{r} v\, dt$$

The power P_wind flowing through the rotor area is then

$$P_\mathrm{wind} \equiv \frac{dE_k}{dt} = \frac{1}{2}\rho A_\mathrm{r} v^3$$

In reality, turbines can only extract a fraction of this available wind power. Betz (1920) demonstrated that at most $16/27 \approx 59.3\%$ of the wind's kinetic energy can be converted into mechanical energy using a wind turbine; real wind turbines never reach this theoretical maximum. The turbine's torque must then be converted to electrical power, leading to further losses.

All in all, a turbine can be characterized by its power curve $P_\mathrm{out}(v)$, indicating the turbine's electrical power output as a function of wind speed (Figure 2). For

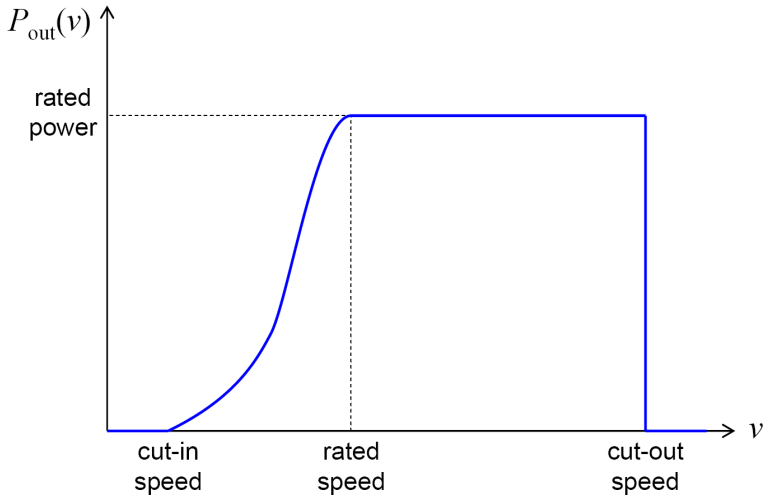


Figure 2: Diagram of a typical power curve.

most common wind turbine designs, $P_\mathrm{out} = 0$ below a cut-in wind speed of $v = 4$ m/s. The power output then rises sharply until v reaches the so-called rated wind speed, usually 12 to 15 m/s; here, P_out plateaus at the rated power (i.e. the nominal power of the wind turbine). The power output is more or less constant until v reaches its cut-out value of 25 m/s (90 km/h), at which point P_out falls to zero as safety mechanisms shut down the turbine to prevent damage. The power curve thus has two regions where the power output varies very rapidly with wind speed. It is therefore important to minimize the error on predicted wind speed, in order to avoid large errors in predicted power output.

Prediction: how?

The first step in predicting the wind resource at a given site, is collecting measurements of wind speed and direction at a nearby location over several years, for example using a meteorological mast. These measurements are compiled into a

description of the wind climate at the mast, usually a probability distribution of the wind speed and direction.

This observed wind climate must then be somehow related to the wind climate at the proposed wind turbine site at hub height. Indeed, the wind climate at the wind turbine is not identical to the wind climate measured at the meteorological mast, due to the different terrain features surrounding these two locations: different obstacles, surface roughnesses, orographies, and thermal effects affect the local wind in different ways at these two points. This is where wind flow modelling comes into play.

Once the wind climate at the hub is predicted, the annual energy production of the projected wind turbine can be estimated with the use of the power curve.

1.3 The WAsP approach

The Wind Atlas Analysis and Application Program (WA^{S}P) was developed as a commercial PC-based tool to estimate the wind resource at a potential wind turbine site, given a set of wind climatology measurements taken at a nearby meteorological mast, and information about the local terrain (Risø National Laboratory 1987). The wind measurements are compiled into a wind rose, which is a histogram showing the directional distribution of the wind, usually divided into 12 bins or sectors. To each 30^o sector is associated a Weibull distribution (Weibull 1951), describing the frequency distribution of wind speed in each sector.

Once these data are input, WA^{S}P employs a two-step approach (Troen and Petersen 1989):

1. The measured wind data is "cleansed" of the effects of local terrain features around the mast, using three models run in succession (described below). The result is a generalized wind climate, or "atlas", that would be observed if, instead of the actual terrain, we had a flat, obstacle-free surface with a uniform, standard roughness length. It is a representation of the regional wind climatology, dependent only on large-scale features, and is thus applicable over a larger area, typically several tens of kilometers.

2. Assuming that this wind atlas is valid over a larger region, the effects of the local terrain features that surround the turbine site are introduced, to obtain the predicted wind climate as seen by the turbine.

Calculations for steps 1 and 2 are performed independently, in two separate micro-scale computational domains, centered on the meteorological mast and the turbine, respectively.

The local terrain features — also known as the *topography* — in both these steps are conceptually divided, according to their size and proximity, into three categories. Each of these is handled by a separate model, as three sub-steps of both steps 1 and 2:

- Discrete **obstacles** in the vicinity of the site (buildings, trees, walls): In the wake of an obstacle, the wind speed is reduced by an empirical factor depending on the dimensions, porosity, and position of the obstacle.

- Surface **roughness** and roughness changes (grass, water surfaces, forest, urban areas): Empirically modelled perturbation of vertical velocity profiles (piece-wise logarithmic)

- **Orography** (hills and valleys):
  - Calculation of the terrain-induced potential flow perturbation
  - Correction for turbulent momentum transfer near the surface

In reality, the perturbations caused by obstacles, roughness changes, and orography interact to produce second-order effects. As a first-order approximations, however, the three calculated perturbations are simply added together.

1.4 Objective: a new orography model for WAsP

The current WAsP orography model is a potential-flow solver (see more detailed review on page 24). It is very quick, and accurate in flat to mildly undulating terrain, and has been used commercially with no inconsiderable success for two decades. However, it performs poorly when applied to complex orography: slopes steeper than about 30% lead to flows that violate its basic assumptions, especially in terms of flow separation, and thus discrepancies arise between the predicted and actual flow perturbations, most notably the speed-up. The speed-up effect occurs at the top of hills, as streamlines are compressed and wind speed is increased; hilltops thus represent an area of interest for the placement of wind turbines.

The present study relates the development of a model for incompressible wind flow better able to accommodate steep slopes, in order to improve wind resource estimation in complex terrain. This model is to be included in a future version of WAsP, complementing or replacing the orography model of the current version.

Requirements

WAsP is a commercial program used by the wind energy community. This poses additional requirements on the new flow model:

Quickness The model must be quick at its user end, so that it can be used on a routine basis on a standard PC platform (as opposed to a research supercomputer) to generate the predicted wind climate at a turbine site within, say, one work day.

Ease of use and stability It must be able to function with very limited user intervention, and must not demand extensive knowledge of Computational Fluid Dynamics on the user's part. The number of user-input parameters must therefore be kept to a minimum. This also means that the model must be stable, to ensure that convergence takes place without the user needing to perform extensive fine-tuning.

From the viewpoint of wind resource assessment, current non-linear flow models based on finite-volume or finite-difference techniques (to be reviewed on page 27) provide very accurate results, notably offering great improvements over WAsP and similar models in the case of steep hills. They are, however, much too expensive computationally to be suitable for routine use on personal computers, and will probably remain so in the coming decade or two. Moreover, the proper use of such models often requires a good deal of fluid dynamics knowledge, lest results of uncertain quality be returned.

There exists a gap, which we have sought to bridge, between models that are fast but insufficiently accurate, and models that are accurate but too slow and complicated to use. To this end, we developed a new flow model, called the Risø Atmospheric Mixed Spectral-Integration Model (RAMSIM), based on first principles and a linear approximation. Before giving a description of our flow model, we turn to an introduction of those principles it is based on, and a review of how they are treated in current scientific and engineering practice.

1.5 Problem definition: The modelling of turbulent flow

The equations describing the motion of fluid, the *Navier-Stokes equations* (Equation 1, page 17), have been known for a very long time: they were formulated in the first half of the nineteenth century. However, even though these equations can be written in a simple and elegant form, they are extremely difficult to solve due to their nonlinearity. In fact, analytical solutions are unobtainable for all but the simplest of flows: laminar (i.e. non-turbulent), steady-state flows in simple, idealized geometries (e.g. flow between two parallel, infinite plates). The phenomenon of *turbulence*, which occurs in the overwhelming majority of flows that surround us, introduces, with its chaotic soup of unpredictable eddies of all sizes, a level of complexity that renders all exact analytical work on the Navier-Stokes equations impossible. The full description of turbulence, for that matter, remains an unsolved riddle to this day, and constitutes a current field of intense research efforts for fluid dynamicists.

Computational Fluid Dynamics

The advent of computers in the second half of the twentieth century has allowed the pursuit of numerical solutions to the Navier-Stokes equations, thus giving birth to an entire field of science, *Computational Fluid Dynamics* (CFD). The bulk of the efforts deployed in this field has been directed at finding approximations and assumptions, tailored to the particular type of flow and geometry studied, that would simplify the Navier-Stokes equations enough for numerical solutions to be feasible with contemporary computers, without sacrificing too much accuracy and detail for the desired application. CFD methods can be classified into three main branches, of which we give a brief overview; for a more complete review, the reader is referred to Ferziger and Perić (2002, chapter 9).

DNS The closest thing to an exception in the quest for simplification is *Direct Numerical Simulation* (DNS), which could be nicknamed the brute-force approach. DNS is an attempt to simulate all the scales of a given flow, from the largest flow features, through the mid-size turbulent fluctuations, down to the smallest turbulent eddies in which kinetic energy dissipates into heat due to friction (molecular viscosity). DNS can thus provide detailed information about the complete structure of a flow. However, since DNS computational grids must be large enough to encompass the entire studied flow, and fine enough to resolve the tiniest details, grid sizes tend to be enormous. DNS is testing the outer limits of currently available computer power (as well as the patience of DNS practitioners): simulations of flows in the simplest of idealized geometries may take many days to complete on the world's most powerful supercomputers. DNS is thus likely to remain a tool of fundamental research in the decades to come, rather than finding any practical application.

LES A step down in detail is *Large-Eddy Simulation* (LES), which uses a coarser computational grid than DNS. The larger eddies are taken to be most important in determining the main features of a flow. Therefore, as the name suggests, the larger eddies are fully simulated, while only the *effect* of the smaller, sub-grid-scale eddies on the larger flow is approximated using models of varying complexities. The unsteady nature of the flow is captured, thus allowing the collection of some turbulence statistics. The computational burden associated to LES, however, is still considerable, and thus LES has been applied mostly to problems involving isolated geometries, such as the flow around an airfoil.

RANS In many applications, the subject of interest is the so-called mean flow, while the moment-to-moment fluctuations due to passing turbulent eddies are considered to be superfluous detail. In the *Reynolds-averaged Navier-Stokes* (RANS) equations, terms that depend on these fluctuations are parameterized in terms of mean-flow variables, so that in the end, only variables describing the mean flow appear (discussed on page 19). The effect of the turbulent eddies on the mean flow is modelled using so-called *turbulence closures* of varying complexities, but individual eddies are not resolved. The level of detail considered, and the associated calculation burden, are generally much reduced; RANS models have thus found a wide range of practical applications.

In our case, as we are interested in predicting the mean output of wind turbines, and not the fine, instantaneous structure of the wind flow, the RANS equations are capable of providing the desired level of detail. Since Reynolds averaging in effect represents a time averaging, however, the effect of time-dependent phenomena such as fluctuating recirculation regions cannot be fully captured. We now turn to a brief discussion of the origins of the RANS equations and of turbulence closure models.

Equations governing the turbulent flow of air in the atmosphere

The atmospheric boundary layer (ABL) can be described as a continuous field of seven variables: $(u_1, u_2, u_3, \rho, P, T, q)$, where u_i is the component of the velocity along coordinate x_i, ρ is the density, P is the pressure, T is the temperature, and q is the moisture content. These variables are related through the following equations:

- The equation of motion (Navier-Stokes equations, relating the change of the three components of velocity to the pressure gradient, friction, thermal buoyancy, and the Coriolis force)
- Continuity equation (conservation of mass)
- Water continuity equation (water is neither created nor destroyed, but may change phase)
- Gas law (relating pressure, temperature, and density)
- Thermodynamic equation (changes of temperature due to heating or cooling, compression or rarefaction, etc.)

In the never-ending quest for completeness, a number of other variables and equations can also be added to this procession. In regions where important amounts of liquid-phase water is suspended, such as fog and clouds, droplet dynamics become significant. Suspended sand, sea salt, and other aerosols also play a role, notably in droplet formation and precipitation. Electromagnetic effects can come into play in clouds, and radiative heat transfer can have an influence in situations where other heat transport mechanisms are suppressed. One has to stop the madness somewhere, however; these effects will not be discussed further.

General equations The Navier-Stokes and continuity equations, expressed in the Earth's rotating frame of reference, and using the Einstein notation, are, respectively:

$$\rho\frac{\partial u_i}{\partial t} + \underbrace{\rho u_j\frac{\partial u_i}{\partial x_j}}_{\text{advection}} = \underbrace{-\frac{\partial P}{\partial x_i}}_{\text{pressure}} + \underbrace{\frac{\partial}{\partial x_j}\left(\mu_{\mathrm{m}}\frac{\partial u_i}{\partial x_j}\right)}_{\text{diffusion}} \underbrace{-2\tilde{\Omega}\epsilon_{ijk}\eta_j u_k}_{\text{Coriolis}} + F_i \tag{1}$$

$$\frac{\partial \rho}{\partial t} + \frac{\partial(\rho u_j)}{\partial x_j} = 0 \tag{2}$$

where x_1 and x_2 are the horizontal coordinates, x_3 is the vertical coordinate, μ_m is the dynamic viscosity of air, η_j is the unit vector parallel to the Earth's axis of rotation, $\tilde{\Omega} = 2\pi/T_\mathrm{rot}$ is the angular frequency of the Earth's rotation, (with $T_\mathrm{rot} = 23$ h 56 min 4 s), and ϵ_{ijk} is the Levi-Civita permutation symbol. The kinematic viscosity (which we will also refer to as the molecular viscosity) is defined as $\nu_\mathrm{m} \equiv \mu_\mathrm{m}/\rho$.

F_i represents the body forces acting on the fluid. The only body force of significance in the ABL is gravity; hence, $F_i = -\rho g \delta_{3i}$, where g is gravitational acceleration and δ_{ji} is the Kronecker delta.

The Coriolis force arises as a side-effect of our using a non-inertial frame of reference, namely the ground, which rotates once daily about the Earth's axis. Seen from this frame, the Coriolis force has the effect of pulling moving objects perpendicularly to their direction of motion, laterally to the left in the Southern hemisphere, and to the right in the Northern hemisphere. The Coriolis force also has a vertical component, causing a small change in the apparent weight of moving objects; this change is much smaller than the gravitational force and is often neglected. For the sake of completeness, we note that in the non-inertial reference frame, the Earth's rotation also results in a small centrifugal force, pulling objects away from the axis of rotation; the corresponding term is customarily absorbed into the gravity force, and as such does not appear explicitly in Equation (1).

Boussinesq approximation The density can be viewed as consisting of a mean part, $\overline{\rho}$, and a fluctuating part, ρ', such that $\rho = \overline{\rho} + \rho'$. The only factor causing significant fluctuations in the density is temperature fluctuations T' about the mean value $\overline{T}$. Since $|T'|/\overline{T}$ rarely exceeds a few percent locally, we can take a first-order approximation of the gas law to yield

$$\rho' \approx -\overline{\rho}\frac{T'}{\overline{T}} \tag{3}$$

The study of the effects of density fluctuations is greatly simplified by the Boussinesq approximation, which postulates that ρ' is too small to cause appreciable differences in inertia, but large enough to cause buoyancy effects. The density is thus treated as a constant ($\rho = \overline{\rho}$) in all terms of Equations (1) and (2) except the gravitational term.

The pressure and gravitational term are treated as

$$-\frac{\partial P}{\partial x_i} - (\overline{\rho} + \rho')g\delta_{3i} = -\frac{\partial}{\partial x_i}(P + \overline{\rho}g x_3) - \rho' g \delta_{3i} = -\frac{\partial p}{\partial x_i} - \underbrace{g\frac{T'}{\overline{T}}\delta_{3i}}_{\text{buoyancy}} \tag{4}$$

where we defined the hydrostatic pressure $p \equiv P + \overline{\rho}g x_3$. The gravitational term has become a buoyancy term, while the hydrostatic balance between gravity is and the vertical pressure gradient is packed into p, to which we will refer simply as "pressure".

Assumptions The present work limits itself to neutrally stable conditions; we thus neglect thermal effects and drop the buoyancy term in the momentum equation. This approximation is by no means generally valid; however it is a necessary first step in the development of our flow model. The treatment of temperature and the transport of thermodynamic energy will be the subject of future investigation.

Humidity also has an effect on the density of air; however, it is usually much smaller than the effect of temperature, and will not be included in our flow model.

For additional simplicity, we chose to neglect the Coriolis force, responsible for the turning of the wind with height, though this effect will be included in future studies.

In summary, we neglect changes in the variables (ρ, T, q) and the equations coupling them to the flow (gas law, water continuity, thermodynamic). We are left with the field variables u_i and p, and their governing equations:

$$\frac{\partial u_i}{\partial t} + \underbrace{u_j \frac{\partial u_i}{\partial x_j}}_{\text{advection}} = \underbrace{\frac{1}{\rho}\frac{\partial p}{\partial x_i}}_{\text{pressure}} + \underbrace{\nu_\text{m} \frac{\partial^2 u_i}{\partial x_j \partial x_j}}_{\text{diffusion}} \tag{5}$$

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{6}$$

Equations for the mean flow

A turbulent flow can be thought of as consisting of a mean flow plus a fluctuating, zero-mean part; this is expressed by the Reynolds decomposition:

$$\left.\begin{aligned} u_i &= \bar{u}_i + u_i' \\ p &= \bar{p} + p' \end{aligned}\right\} \tag{7}$$

with

$$\overline{u_i'} = 0$$

$$\overline{p'} = 0$$

The $\overline{\text{overline}}$ denotes ensemble averaging. In practical, intuitive terms, this can be thought of as time averaging over an interval sufficiently long to average out the rapid turbulent fluctuations, yet short enough not to blur out the broad variations in the mean flow.

In the context of wind energy resource assessment, we are interested in the characteristics of the mean flow, and not so much in the details of the rapidly fluctuating part. Equations for the mean flow are obtained by substituting Equation (7) into Equations (5) and (6) and ensemble-averaging. This approach, put forth by Reynolds in 1895, yields:

$$\frac{\partial \bar{u}_i}{\partial t} + \bar{u}_j \frac{\partial \bar{u}_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial \bar{p}}{\partial x_i} + \frac{\partial}{\partial \acute{x}_j}\nu_\text{m}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) - \frac{\partial \tau_{ij}}{\partial \acute{x}_j} \tag{8}$$

where the Reynolds stress tensor τ_{ij} is defined as

$$\tau_{ij} = \overline{u_i u_j} - \bar{u}_i \bar{u}_j = \overline{u_i' u_j'}$$

The closure problem

The Reynolds stresses are additional unknowns, which were not present in the original equations for turbulent flow (5). One may attempt to obtain expressions for these second-order correlations by deriving an equation for the fluctuating part of the flow; this is done by subtracting Equation (8) (mean flow) from Equation (5) (total flow), and using the continuity equation to rearrange. Doing this, however, generates yet more unknowns, in the form of third-order correlations of the type $\overline{u_p' u_q' u_r'}$. Further attempts to obtain expression for these are futile; it can indeed be shown that the derivation of an equation for an unknown n^th-order correlation will generate ever more extra unknowns in the form of $(n+1)^\text{th}$-order correlations. The equations for mean turbulent flow can thus not be solved unless additional assumptions are made. This is known as the closure problem.

Turbulence closure models

The full effects of turbulence are thus impossible to calculate using the RANS equations; one must resort to closure models that are built on certain assumptions. Many different turbulence closure models have been developed, in which the effects of turbulence are modelled with varying degrees of complexity and physical completeness.

The choice of turbulence closure can have a tremendous impact on the calculated turbulence field, but has a lesser influence over the mean velocity field, at least when there is no flow separation, as documented by Taylor, Mason and Bradley (1987), Ayotte, Xu and Taylor (1994) and Castelli, Ferrero and Anfossi (2001), and perhaps also in the case of flow separation (Kim and Patel 2000). An accurate representation of the turbulence field is of great importance when making dispersion calculations or estimating peak wind loads. However the main concern of the present work is the calculation of the mean velocity field, which is not quite as sensitive to the choice of turbulence closure, and thus we are at greater liberty to shun the most complex closures and their associated greater computational load.

Parameterization of the Reynolds stresses

One approach to the closure problem is to interpret the Reynolds stresses as a turbulent addition to the molecular stresses. Intuitively speaking, these two stresses act in a similar way: both have a diffusive character, in that they contribute to the local "mixing" of fluid properties, momentum in particular, with the net effect of transporting momentum (and other properties) against the local gradient.

As noted by Wilcox (1998, chapter 3), the theoretical basis of this analogy is somewhat shaky, as it does not rest on fully rigourous physics. It has nevertheless found widespread application, with no inconsiderable success at describing flows in practice. It can be expressed mathematically in the form of the Boussinesq eddy-viscosity approximation, which relates the Reynolds stresses to the gradient in the mean-flow velocity.

The Reynolds stress tensor τ_{ij} is symmetric ($\tau_{ij} = \tau_{ji}$) but not traceless ($\tau_{ii} \neq 0$), while $\partial u_i/\partial x_j$ is traceless ($\partial u_i/\partial x_i = 0$) but not symmetric. The two are related by a turbulent transfer coefficient called the eddy viscosity ν_t by postulating

$$\tau_{ij} - \frac{1}{3}\tau_{kk}\delta_{ij} = \nu_\mathrm{t}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)$$

which makes both sides symmetric and traceless. By defining the turbulent kinetic per unit mass (henceforth abbreviated TKE), E, as half of the averaged sum of the squared velocity fluctuations,

$$E = \frac{1}{2}\tau_{kk} = \frac{1}{2}\overline{u'^2} \tag{9}$$

we can write

$$\tau_{ij} = \nu_\mathrm{t}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) - \frac{2}{3}E\delta_{ij} \tag{10}$$

We thus introduce a "total" kinematic viscosity ν consisting of a molecular part and a turbulent part (eddy viscosity):

$$\nu = \nu_\mathrm{m} + \nu_\mathrm{t} \tag{11}$$

In practice, when turbulence is fully developed, as it is over most of the atmospheric flow field, $\nu_\mathrm{t} \gg \nu_\mathrm{m}$, such that we can reasonably approximate $\nu \approx \nu_\mathrm{t}$ — and we do so in the remainder of this document. This approximation does not hold very near the surface, where the flow is highly viscous in character; however, the present work is not concerned with the study of flow in this region.

In much of the geophysics literature, ν_t is represented by the letter K, and the set of concepts presented here is known as K theory. The earliest and simplest models (such as the 1905 model of Swedish oceanographer V. W. Ekman) simply assumed a constant K. This approximation is sufficient when seeking estimates of small-scale diffusion of passive tracers in the high-altitude free atmosphere, where turbulence has fairly uniform characteristics. However, it is not realistic in the boundary layer, where turbulent eddies vary strongly in size and intensity with increasing distance from the ground. How does one then go about obtaining an expression for ν_t?

One-equation models

Arguably the simplest way in which one can characterize turbulence is by its TKE and a length scale, the mixing length l_{mix}. Equivalently, a velocity scale $q = \sqrt{2E}$ can replace the TKE. Dimensional arguments lead to the expression

$$\nu_t = C_\mu q l_{\text{mix}} \tag{12}$$

where C_μ is a constant. Mixing-length models employ this expression, equating q to the local velocity gradient times l_{mix}. In simple geometries, the mixing length l_{mix} can be proportional to, for example, the distance from a surface. Such a length scale is, however, virtually impossible to prescribe in the case of three-dimensional flows, separated flows, or complex geometries.

Two-equation models

In equilibrium turbulent flows, the rates of production of TKE, Π, and of its dissipation into heat due to viscosity effects, ε, are in near-balance. The dissipation can be thought of as representing

$$\varepsilon_{ij} = \overline{\nu_m \frac{\partial u_i'}{\partial x_q}\left(\frac{\partial u_j'}{\partial x_q} + \frac{\partial u_q'}{\partial x_j}\right)} \tag{13}$$

where, under the assumption of isotropy (identical properties in all directions),

$$\varepsilon_{ij} = \frac{1}{3}\varepsilon\delta_{ij}$$

In this case, the following relationship can be used:

$$\varepsilon = C_{\text{owl}}\frac{E^{\frac{3}{2}}}{l_{\text{mix}}} \tag{14}$$

where C_{owl} is a constant. This result can be derived based on the concept of the energy cascade, whereby energy is continuously transferred to eddies of smaller and smaller scales (and eventually dissipated at the Kolmogoroff scale), or simply on dimensional arguments.

This leads to the expression

$$\nu_t = C_\mu \frac{E^2}{\varepsilon} \tag{15}$$

where the constant C_{owl} was absorbed into C_μ.

The $E-\varepsilon$ model (also known as $k-\varepsilon$ in the engineering literature) was developed by Jones and Launder (1972) and Launder and Sharma (1974) based on these considerations. It has become the most popular two-equation turbulence closure, and will be discussed here. A number of other two-equations closures have also been developed, notably the $E-\omega$ model, where ω is a turbulence frequency scale defined by the specific dissipation of TKE, ε/E. For a review of this and other two-equation closures, the reader is directed to Wilcox (1998, chapter 4).

Standard $E-\varepsilon$ model An equation for E can be inferred from the Navier-Stokes equations, by treating the TKE as a transported quantity:

$$\frac{\partial E}{\partial t} + \bar{u}_j \frac{\partial E}{\partial x_j} = \underbrace{\frac{\partial}{\partial x_j}\left(\frac{\nu_\mathrm{t}}{\sigma_E}\frac{\partial E}{\partial x_j}\right)}_{\text{diffusion}} + \Pi - \varepsilon \tag{16}$$

where Π and ε are source terms that represent the rate of production and destruction (or dissipation) of TKE, respectively. TKE is "produced" when kinetic energy "dissipates" from the mean flow and is transferred to the turbulent eddies, and "destroyed" when it dissipates into heat at the level of the smallest eddies due to molecular viscosity. The production of TKE stems from mean-flow dissipation, which in turn is modelled to be analogous to viscous dissipation into heat. We thus obtain an expression for the production Π that is analogous to Equation (13):

$$\begin{aligned}\Pi &= \nu_\mathrm{t}\frac{\partial \bar{u}_i}{\partial x_j}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right) \\ &= \frac{\nu_\mathrm{t}}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)^2\end{aligned} \tag{17}$$

A transport equation for the dissipation ε can also be modelled on the Navier-Stokes equations, and parameterized as

$$\frac{\partial \varepsilon}{\partial t} + \bar{u}_j \frac{\partial \varepsilon}{\partial x_j} = \underbrace{\frac{\partial}{\partial x_j}\left(\frac{\nu_\mathrm{t}}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x_j}\right)}_{\text{diffusion}} + \underbrace{\frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right)}_{\text{source term}} \tag{18}$$

The source term models how dissipation is "produced" and "destroyed" — somewhat artificial concepts. The production and destruction of ε are made to follow those of E, on the intuitive grounds that the more TKE is made available, the more dissipation into heat will take place, and vice-versa; but they are scaled by the empirically-determined constants $C_{\varepsilon 1}$ and $C_{\varepsilon 2}$. The empirical constants σ_E and σ_ε similarly scale how E and ε diffuse (see diffusion terms of Equations 16 and 18). Note that the ε equation (18) is not based on entirely rigourous physics, and may not represent the "true dissipation" very well; however, it has proven quite successful at yielding the appropriate ν_t to model a variety of flows correctly, and is therefore widely used based on this empirical success. The model constants prescribed by Launder and Spalding (1974) are listed in Table 1.

Table 1: Standard E-ε model constants

constant	C_μ	σ_E	σ_ε	$C_{\varepsilon 1}$	$C_{\varepsilon 2}$
value	0.09	1.0	1.3	1.44	1.92

Thus Equations (16) and (18) and the constants listed in Table 1 define the $E-\varepsilon$ closure; this is the turbulence closure we choose to use in our flow model. We turn, however, to a short discussion of one additional two-equation closure of future relevance to the flow model described in this work, actually a modification of the $E-\varepsilon$ closure.

Limited-length-scale $E-\varepsilon$ closure In neutral atmospheric flows, the size of turbulent eddies cannot grow indefinitely, but is rather limited by the depth of the boundary-layer. The standard $E-\varepsilon$ closure, however, leads to a length scale that

grows linearly with height: from Equations (14) and (15), we see that the length scale is related to the dissipation by

$$l_{\mathrm{mix}} = C_\mu^{\frac{3}{4}} \frac{E^{\frac{3}{2}}}{\varepsilon} \tag{19}$$

and the dissipation, in an equilibrium shear flow, is inversely proportional to height (see Equation 75 in Section 3). The mixing length scale is therefore

$$l_{\mathrm{mix}} = \kappa z$$

Applied to boundary-layer flows, the standard $E-\varepsilon$ model thus gives inordinately large mixing lengths (and thus eddy viscosities) as z increases, and exaggerates the deepness of the boundary layer. This was observed by Detering and Etling (1985) upon comparing $E-\varepsilon$ calculations with data from the Leipzig wind profiles. In contrast, they found that one-equations models performed well when the mixing length was limited algebraically as

$$l_{\mathrm{mix}} = \left(\frac{1}{\kappa z} + \frac{1}{l_{\max}} \right)^{-1}$$

given a judicious choice of maximum mixing length $l_{\max}$, e.g. for neutral conditions:

$$l_{\max} = \frac{h_{\mathrm{BL}}}{3} \tag{20}$$

where h_{BL} is the height of the boundary layer. This height may be estimated from surface data as $h_{\mathrm{BL}} \approx 0.2\, U_* f_{\mathrm{c}}$, where U_* is the friction velocity and f_{c} is the Coriolis parameter.

Apsley and Castro (1997) addressed these deficiencies of the standard $E-\varepsilon$ closure by developing a modified $E-\varepsilon$ closure that reflects the fact that the growth of eddies is capped by the height of the boundary layer. They achieved a limitation of the mixing length by modifying the ε production term, such that the ε transport equation becomes

$$\begin{aligned} \frac{\partial \varepsilon}{\partial t} + \bar{u}_j \frac{\partial \varepsilon}{\partial x_j} &= \frac{\partial}{\partial x_j} \left(\frac{\nu_{\mathrm{t}}}{\sigma_\varepsilon} \frac{\partial \varepsilon}{\partial x_j} \right) \\ &\quad + \left(C_{\varepsilon 1} + (C_{\varepsilon 2} - C_{\varepsilon 1}) \frac{l_{\mathrm{mix}}}{l_{\max}} \right) \frac{\varepsilon}{E} \Pi - C_{\varepsilon 2} \frac{\varepsilon^2}{E} \end{aligned} \tag{21}$$

As l_{mix} increases toward $l_{\max}$ with increasing height, the production of ε increases (since $C_{\varepsilon 2} > C_{\varepsilon 1}$), and the increased ε thus acts as a negative feedback on the mixing length defined in Equation (19).

This limited-length-scale $E-\varepsilon$ closure has proven more successful at modelling boundary-layer wind profiles than the standard $E-\varepsilon$, yielding realistic boundary-layer depths as well as turning of the wind with height. We confirmed this by performing, using our prototype flow model (see Section 1.7, page 30), a preliminary simulation of the famous Leipzig vertical wind profile, measured over flat terrain in conditions of near-neutral thermal stratification (Lettau 1962). We discretized the equations (including the Coriolis term) for horizontally homogeneous flow over flat terrain in a finite-difference formulation and solved them by iteration, with a moving-lid upper boundary and a law-of-the-wall, rough lower boundary. Results using the standard and limited-length-scale $E-\varepsilon$ closures are compared on Figure 3.

For the sake of simplicity, the current version of our flow model does not include the Coriolis force, and thus assumes an infinitely deep boundary layer. As we will see in Section 3, this allows us to use analytical expressions for the unperturbed (zero-order) flow; we therefore employ the standard $E-\varepsilon$ closure. Future versions of our flow model will remove this limitation and employ the limited-length-scale $E-\varepsilon$ closure.

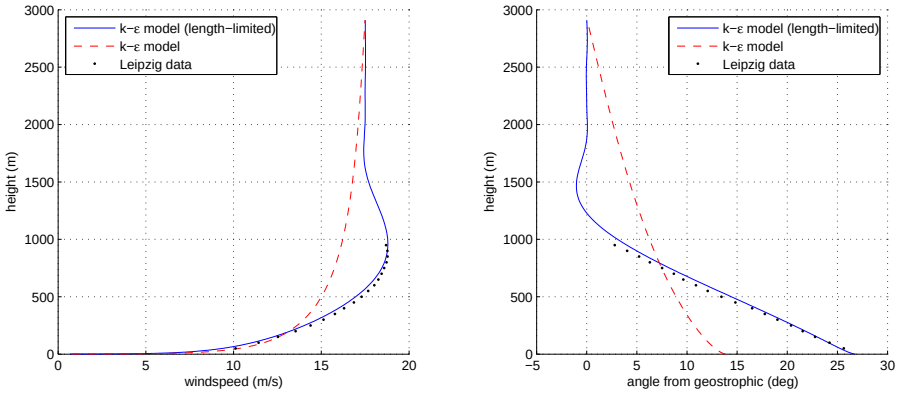


Figure 3: Vertical profiles of the wind speed (left) and direction (right): Leipzig data compared to calculations using the standard and limited-length-scale $E - \varepsilon$ closures.

Higher-order closures

The one- and two-equation closures discussed above rely on the Boussinesq eddy-viscosity approximation, by which the Reynolds stress tensor τ_{ij} and the mean strain rate tensor,

$$S_{ij} = \frac{1}{2}\left(\frac{\partial \bar{u}_i}{\partial x_j} + \frac{\partial \bar{u}_j}{\partial x_i}\right)$$

are related by a simple proportionality constant, the eddy viscosity ν_t. In certain flows, however, the Reynolds stresses are significantly affected by flow-history effects, and cannot be assumed to have a linear relationship to mean-flow features. This is the case for flows with sudden changes in mean strain rate, or with inordinately large strain rates, such as those caused by extreme streamline curvature. These deficiencies have been addressed via two basic approaches.

- **Nonlinear constitutive relations** The Boussinesq eddy-viscosity approximation, defined by Equation (10), can be viewed as the first term in a series expansion, to which higher-order, non-linear terms can be added, following different assumptions.
- **Stress-transport models** A transport equation for τ_{ij} can be derived by multiplying the Navier-Stokes equation by the fluctuating u_i' and ensemble-averaging the result. One thus obtains six new equations (accounting for the symmetry of τ_{ij}), but also twenty-two new unknowns (higher-order correlations) as a by-product. These unknowns must in turn be accounted for by new closure approximations, of which a number have been developed, as reviewed by Wilcox (1998).

Neither of these approaches are trivial, and both entail, upon application, a significant increase in computational burden. Since good results for flow over orography have previously been obtained using the simpler $E - \varepsilon$ closure, including in the occurrence of flow separation (Jørgensen et al. 2006), we deem these higher-order closures too complex for our purposes at this point.

1.6 Wind flow over orography: a review

Over the past three decades, our understanding of the effects of topography on ABL flow has improved substantially as a result of extensive work on the theoretical, numerical modelling, and experimental fronts. This progress has been

reviewed by Kaimal and Finnigan (1994) and Belcher and Hunt (1998), among others.

The next few pages are devoted to a brief review of the modelling efforts of the past 30 years. A few wind tunnel and field experiments will be reviewed in Section 6, where their results are compared to the output of our flow model.

Early analytical work Early work focused on developing an analytical model of the flow perturbation over hills. Due to the intractability of the RANS equations, which describe the mean flow of air, analytical work has only been possible on a simplified, linearized version of these equations, and is therefore known as *linear theory*. Typically, the ABL is conceptually divided into two to four layers. In each of these layers, the flow is argued to be dominated by different effects; the equation terms representing the other, locally less important effects are simplified or neglected, leading to equations of more manageable complexity — typically a balance between two dominant terms. These equations can be solved analytically, and may be matched at the interfaces between the various layers and to boundary conditions.

Jackson and Hunt (1975) (JH) were the first to develop an analytical model for two-dimensional wind flow over a low hill. This model provided the basis for subsequent efforts by a number of authors; we will thus give a short overview of its substance.

Jackson and Hunt considered the asymptotic case where the surface roughness z_0 is uniform and negligible relative to L, the horizontal length scale of the hill, and where there is no mean flow separation. The flow is considered to consist of the basic, upstream flow $u_0(z)$ over flat terrain, displaced upward by the shape of the hill, plus a perturbation Δu introduced by the effect of flowing over the hill (Figure 4). It is formally assumed that $\Delta u \ll u_0$, though the theory often still holds even though this assumption is not strictly fulfilled in practice.

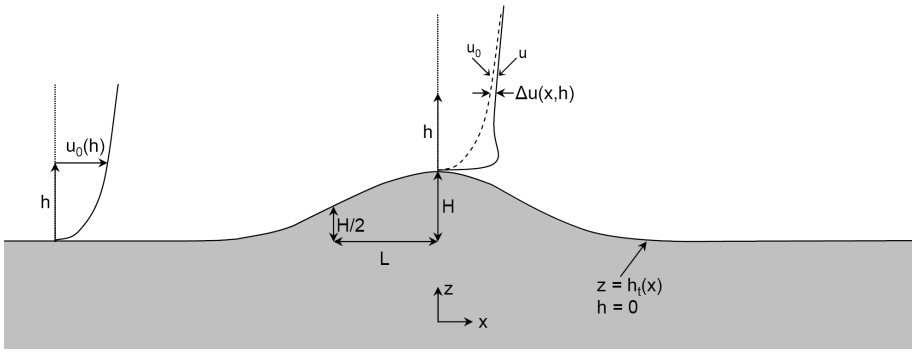


Figure 4: Diagram of the flow over a hill. H is the hill height, L the hill length scale (half-width at half-maximum), h_t the terrain height, h the height above ground level.

Jackson and Hunt divided the flow vertically into two layers, inner and outer, that are demarcated at a height of order l. In the outer layer (above l), the flow is considered to be inviscid, the basic flow is constant with height (since $z_0 \to 0$), and there are significant pressure perturbations, decreasing with height. In the inner layer (below l), the basic flow is logarithmic (but is assumed to be constant with height in the advection term), and there are significant shear stress perturbations.

The magnitude of l is derived from an order-of-magnitude balance between the advection and stress divergence terms in the equation of momentum transport,

yielding

$$\frac{l}{L}\ln\left(\frac{l}{z_0}\right) = 2\kappa^2$$

where $\kappa = 0.4$ is the von Kármán constant.

Turbulent transport is represented by a mixing-length model. The equations are Fourier-transformed horizontally, and an analytical solution is obtained in which the perturbation increases with height. The JH model provided predictions for the mean wind speed changes and turbulence structure, but was strictly limited to hills with small slope, i.e. $H \ll L$, where H is the hill height.

In the following years, a number of authors worked on removing some of the limitations of the JH model. Mason and Sykes (1979) extended the JH model to three-dimensional hills (this extension is commonly called MS3DJH), and obtained reasonable agreement with observations when applied to a real hill of moderate slope.

Sykes (1980) improved linear theory by introducing a thin wall layer between the ground surface and the inner layer, providing a better match to the lower boundary condition. Hunt, Leibovich and Richards (1988) subdivided the outer layer into two sublayers: a middle region, where the flow is effectively inviscid, but still under the influence of mean vorticity; and a redefined outer region, in which the flow is taken to be inviscid and irrotational.

Walmsley, Salmon and Taylor (1982) addressed the unrealistic propagation to high altitudes, inherent to previous applications of linear theory, of high-wavenumber effects caused by small features in real terrain. This was due to the assumption that the pressure perturbation field was independent of height. In their modification, Walmsley et al. changed the vertical coordinate used in the outer layer, and found that the solution to the resulting pressure perturbation equation declined with height in a wavenumber-dependent fashion. Using this new pressure perturbation in the inner-layer equations led to a considerable reduction in high-wavenumber noise at high elevations.

The current WA^{S}P model for wind flow over orography, developed by Troen and de Baas (1986) and Troen and Petersen (1989) at Risø National Laboratory, has many similarities with the family of models based on the JH model. In contrast to its predecessors, however, it is tailored to the task of calculating the velocity perturbation at a single point of interest, namely the location of a meteorological mast or projected wind turbine, placed at the center of a polar computational grid. For any given wind speed and direction, the velocity perturbation $\Delta\vec{u}$ is related to its potential χ by $\Delta\vec{u} = \nabla\chi$. The potential is decomposed into a Bessel-Fourier expansion, whose coefficients are calculated to satisfy the surface kinematic boundary condition (continuity near the surface). The velocity perturbation at a height h above ground is then calculated from each Bessel-Fourier component i as

$$\frac{\Delta\vec{u}_i(h)}{u_0(h)} = \frac{|u_0(L_i^{\text{out}})|^2}{\left|u_0[max(h, L_i^{\text{in}})]\right|^2}$$

where $u_0(h)$ is the upstream wind profile, and L^{in} and L^{out} are inner- and outer-layer length scales, analogous to the l and L length scales of the JH model. Since χ is calculated independently at each point of interest, and not on a complete Cartesian grid (as is commonly done in other models), WA^{S}P is very quick at calculating the flow at a number of discrete, projected wind turbine sites, but is less efficient at producing gridded output, such as a high-resolution map of the wind resource over a given area.

The LINCOM model, also developed at Risø, was derived from the WA^{S}P model in order to be included in the WA^{S}P Engineering program (Risø National Laboratory 1999), a complementary tool for the estimation of extreme winds, shears, flow angles, wind profiles, and turbulence over complex terrain. Contrary

to its parent, LINCOM calculates the wind vector in every mesh point of a rectangular grid, and uses a different model for the influence of variable surface roughness (Astrup, Mikkelsen and Jensen 1997), including the wind-speed-dependent roughness of the sea surface. LINCOM uses a simple one-equation turbulence closure.

Linear theory was applied extensively in the 1970s and 1980s due to its low requirements for computer processing and memory, but models derived from it have also had continued influence up to the present, because it gives reasonably good results in the upstream region and on the top of hills. However, it suffers from two major limitations. First, its validity is formally limited to small flow perturbations, i.e. shallow hills, due to the underlying linear approximation. Second, all the linear models listed above used a mixing-length closure for the turbulent stress perturbations; more complex turbulence closures were prohibitively difficult to include in the analytical formulation of the vertical variations in the flow. (A related limitation is the approximation of the non-linear advection term.)

Zeman and Jensen (1987) devised a method for including a more complex turbulence closure by adopting a streamline coordinate system. They assumed a pressure perturbation given by a linearized potential solution based on a given upstream, undisturbed wind profile; this solution was obtained by numerically integrating a known formula involving a function representing the shape of the hill. Mean flow streamlines were then found by applying the von Mises transformation to the steady-state momentum transport equation (i.e. changing the vertical coordinate from z to the streamfunction ψ), which is then solved by marching along the constant-streamfunction coordinate. The Reynolds stresses are obtained from a linearized second-order turbulence closure. Good agreement was obtained between the model results and measurements taken over Askervein hill (the Askervein experiment will be reviewed in Section 6.4).

Non-linear CFD: finite-volume and finite-difference Attempts were made in the late 1970s to apply non-linear computational fluid dynamics (CFD) models to the flow of wind over orography, but were severely limited by the computer power available at the time.

Taylor and Gent (1974) and Frost, Maus and Fichtl (1974) were the first to attempt the study of wind flow over hills with non-linear finite-difference models. Taylor (1977) successfully developed a finite-difference model for two-dimensional flow over idealized terrain features of moderate slope, but recognized that three-dimensional calculations were not yet feasible.

As computer power increased in the 1980s, CFD calculations over three-dimensional orography became possible. However, the largest grids manageable at the time remained coarse by today's standards, causing a loss of detail in the topography as well as larger spatial discretization errors. This made it difficult to identify and interpret the model limitations caused by other factors. Raithby, Stubley and Taylor (1987) used finite-volume techniques to simulate the flow over Askervein hill in a grid of $20 \times 20 \times 19$ cells.

Wood (1995) performed CFD simulations of the flow over infinite series of two-dimensional sinusoidal hills as well as infinite arrays of three-dimensional, axisymmetric hills, using a $1\frac{1}{2}$-order turbulence closure. By studying the simulated flow over hills of varying amplitude, with maximum slopes ranging from 0.16 to 0.94, Wood derived an analytical expression for the critical slope at which separation occurs, and showed its agreement with some experimental data.

Brown and Wood (2001) simulated the neutrally-stratified flow over an infinite array of ellipsoidal hills using a three-dimensional finite-difference model developed earlier (Wood and Mason 1993) and a first-order mixing-length turbulence closure. They performed numerous simulations, varying both the aspect ratio of the hills and the direction of the wind relative to the hills' axes, and were thus able to

parameterize the components of the pressure drag along the major and minor axes of the hills.

Castro, Palma and Silva Lopez (2003) simulated the flow over Askervein hill with a finite-volume CFD code employing the $E-\varepsilon$ turbulence closure. They performed simulations using grids ranging between $27 \times 27 \times 15$ and $155 \times 155 \times 31$ cells, and found that results depended on the grid resolution in the lee of the hill, where some recirculation was predicted, but not above the upwind slope or hill crest. More recently, Undheim, Andersson and Berge (2006) simulated the Askervein flow using a similar model, but with a variant of the $E-\varepsilon$ closure, whose constants are adjusted as algebraic functions of the local mean strain rate according to renormalization group theory (RNG), as reported by Kim and Patel (2000) and Kim, Patel and Lee (2000). They obtained excellent agreement with experimental data, but reported some dependence of results on the vertical resolution.

A three-dimensional finite-volume code (EllipSys) with a $E-\varepsilon$ closure was developed at Risø National Laboratory by Sørensen (1995). It included two versions of the logarithmic law-of-the-wall boundary conditions (for smooth and rough walls) so as to be able to handle both industrial and atmospheric flows. EllipSys was used to successfully simulate the flow over a variety of hills, including Hjardemål escarpment (Jørgensen, Ott, Sørensen, Mann and Badger 2006), discussed in more detail in Section 6.3.

Eidsvik (2005) used down-scaling of large-scale weather prediction model results to calculate the micro-scale flow in hilly terrain. The results of the HIRLAM hydrostatic weather prediction model (Errico and Baumhefner 1987) provided the boundary conditions for the MC2 meso-scale model (Benoit et al. 1997), whose results in turn provided the boundary conditions, most notably the inflow velocity and turbulence characteristics, for a micro-scale finite-element flow model. The entire system could be forced with historical large-scale meteorological data, and each successive model then focused on a smaller area, and had a finer resolution (10 km, 1 km and 100 m, respectively) as well as a more complete turbulence closure (algebraic mixing-length closure, one-equation closure, and $E-\varepsilon$ closure, respectively) than the previous model. Using this scheme, Eidsvik obtained excellent agreement with field data from Askervein hill, and projected that computing times of a few weeks on a 25-processor computer would be require to simulate a full year's historical data.

Mixed models Full finite-difference models are computationally costly, while the linear analytical models discussed earlier are rigid, in that more complex elements, such as more physically complete turbulence closures or non-linear terms, are prohibitively difficult to include. Beljaars, Walmsley and Taylor (1987) relaxed this rigidity by discretizing the linearized equations and solving them by finite difference vertically, thus introducing a new type of model termed *Mixed Spectral Finite-Difference* (MSFD). MSFD models inherit the reduced computational load of spectral methods, while retaining some of the flexibility of finite-difference models. Beljaars et al. were thus able to use the $E-\varepsilon$ closure, while Ayotte et al. (1994) tested MSFD with a number of different closures, including two more complex second-order closures.

Ayotte and Taylor (1995) relaxed the assumptions on the upstream basic flow in the MSFD model. Their model calculated the upstream flow by iteration using the full, non-linear RANS equations and a second-order turbulence closure over flat terrain, and including the Coriolis force. Nonlinear terms were still neglected when calculating the perturbation.

The linear MSFD model of Beljaars et al. (1987), like other linear models, was reasonably successful at modelling the flow over isolated hills, even in the case of moderately steep slopes apparently violating the linearity assumption. However,

they remained inaccurate in the lee region, where the perturbation (speed reduction, let alone flow separation) is grossly underestimated, and to a lesser extent over hill crests, where the perturbation (including speed-up) is exaggerated close to the ground.

In response to this, Xu and Taylor (1992) and Xu, Ayotte and Taylor (1994) developed NLMSFD, an extension to MSFD that included some non-linear effects, by retaining non-linear terms as extra source terms in the finite-difference calculations. The equations were then solved iteratively in Fourier space, with the non-linear terms, calculated in physical space, lagging behind by one time step. On the one hand, this greatly improved the results in the lee of hills, including in certain cases of flow separation. On the other hand, NLMSFD experienced numerical stability problems, failing to converge for hills with moderate slopes (>0.3) — this was rather unfortunate, since the calculation of the flow over steep hills was precisely the stated aim of the authors' efforts. Taylor (1998) reported that satisfactory NLMSFD simulations of the flow over Askervein hill had still not been obtained.

Large Eddy Simulation Since turbulent eddies evolve in three dimensions, LES requires a three-dimensional grid to correctly represent the turbulence dynamics, even in two-dimensional problems; and this grid must be isotropic so as not to force eddies into artificial structures. Thus fine resolution in one direction implies fine resolution in all three directions, and the number of grid cells grows cubically with resolution.

The first endeavours to apply LES to flow over idealized hills were made in the mid-1990s, but were hampered by the tremendous computational resources required. Despite the increase in computing power in the past ten years, this still represents a major hurdle the application of LES to flow over realistic orography. Another concern is the sensitivity of LES to a variety of parameters, such as the details of the inflow and boundary conditions.

Gong, Taylor and Dörnback (1996) attempted to simulate the boundary-layer flow over a pair of sinusoidal hills, but realized that their grid resolution ($128 \times 128 \times 64$) was inadequate to resolve the inner region. They estimated that the actual number of grid points required would be about two orders of magnitude greater, an estimate later corroborated by Wood (2000).

Brown, Hobson and Wood (2001) successfully simulated the neutral boundary-layer flow over sinusoidal hills with a maximum slope $m = 0.2$. By using finer and finer grids, they demonstrated grid independence over sinusoidal hills with $m = 0.036$, but were logistically prohibited from doing so over the steeper hills, since these required longer convergence times.

Allen and Brown (2002) performed LES simulations of the separated flow occurring behind over rough hills, comparing results with laboratory measurements over an isolated hill and a periodic sinusoidal hill. They reported mixed results, with excellent agreement in the former case and some wake-region discrepancy in the latter. According to the authors, this was due the sensitivity of their simulations to the details of the canopy model used to obtaining a smooth match between the well-resolved flow interior and heights comparable to the size of the roughness elements. They nevertheless demonstrated how much interesting information can be obtained from LES on the turbulent structures and the time evolution of the flow, notably the length of the separation bubble. Ding and Street (2003) performed further LES numerical experiments to study the wake structure behind an axisymmetric hill under varying degrees of thermal stratification.

The first beachheads for LES simulation of the flow over complex terrain are currently being established. Uchida and Ohya (2003) applied LES codes to the calculation of turbulent airflow over an area of 9.5 km $\times$ 5 km covering the new

campus of Kyushu University, Japan, with a horizontal resolution of 50 m and 61 vertical levels. The same group reports using their LES code to simulate the wind resource at the site of an actual wind farm located in complex terrain (Uchida and Ohya 2006). Whether the greater computational expense associated to LES compared to RANS codes is justified by significantly better wind resource predictions remains to be determined.

1.7 The new WAsP flow model: a summary

History and candidates considered

All the work described in the present thesis was carried out in the Wind Energy Department of Risø National Laboratory. The New WAsP project was initiated in early 2002 with a search for possible model types to be considered. Out of many different possible approaches, which one to choose?

The speed requirement provided the answer. Full CFD simulations of wind flow using finite-volume or finite-difference techniques are the state of the art and can give very accurate results. They are, however, very computationally expensive, and will likely remain beyond the horizon of routine use on PCs for at least a decade or two.

We therefore chose to take a step back and further explore the avenue of perturbation solvers, in which the flow over orography is thought of as a basic part equivalent to flow over flat terrain, plus a perturbation part introduced by the presence of the orography. In this we were inspired by the literature of the past three decades.

Prototype Prior to to the development of the current model, a prototype based on the perturbation approach was developed, much inspired by the the MSFD model of Ayotte and Taylor (1995) and the underlying work of Beljaars et al. (1987). Unlike these previous models, however, the prototype used general curvilinear coordinates, and a strong-conservation, invariant tensor form of the governing equations. A hyperbolic grid generator was used to create terrain-following coordinate systems. Apsley and Castro's (1997) limited length-scale $E-\varepsilon$ closure was employed. The basic, unperturbed flow was calculated by solving the non-linear equations for horizontally homogeneous flow by finite-difference. The first-order perturbation terms were calculated numerically by pseudo-spectral methods, and solved vertically by finite-difference, with the use of the fractional step method described by Perot (1993). Progress is recorded in technical reports by Jørgensen (2003b), Jørgensen (2003a) and Myllerup, Corbett and Jørgensen (2005).

After three years of work, however, we abandoned the prototype in February, 2006, due to disappointing and unpromising behaviour, especially in terms of numerical stability (hardly any useable results could be produced), but also of accuracy and computational load.

Current model The flow model described in the present thesis was recently named Risø Atmospheric Mixed Spectral-Integration Model (RAMSIM). Like the previous prototype, it was inspired by the MSFD family of models initiated by Beljaars et al. (1987), but it departs radically from its predecessor in two important respects: 1) The terrain-following transformed coordinate system is not created by a numerical grid generator, but rather based on an *analytical* expression, which can be "built into" the model equations. 2) The solution method in the vertical direction is different: the equations are not discretized and solved by iterative finite-difference methods, but rather rearranged into a set of ordinary differential equations (ODEs), solved by numerical integration.

We coded our numerical model in the FORTRAN 90 programming language, on a standard PC platform. We now turn to a brief overview of the RAMSIM flow model.

Current limitation to 2D

RAMSIM has been implemented and tested in two-dimensional space. However, the derivations for the full 3D model equations is presented in this document. The differences between the 2D and 3D cases will be noted in the text.

Governing equations

RAMSIM is governed by the following equations, expressed in general curvilinear coordinates:

- Steady-state Reynolds-averaged Navier-Stokes (RANS) equations for incompressible flow and neutral thermal stratification:
  - transport equation for momentum
  - transport equation for mass (continuity equation)
- Turbulence closure: The turbulent stresses are parameterized using the $E-\varepsilon$ model. This consists of two equations:
  - transport equation for the turbulent kinetic energy (TKE)
  - transport equation for the dissipation of TKE.

Coordinate system

A terrain-following, transformed coordinate system is created from a simple analytical expression in the Fourier domain. The many factors that arise in the equations due to the coordinate transformation can therefore be replaced by analytical expressions. This is a key feature contributing to the rapidity of RAMSIM.

In the remainder of this document, unless otherwise specified, "horizontal" and "vertical" are to be respectively understood as horizontal and vertical in the transformed coordinate system, where "horizontal" denotes the two terrain-following coordinates, $(\acute{x}_1, \acute{x}_2) = (\acute{x}, \acute{y})$, and "vertical", the coordinate running from the ground to the sky, $\acute{x}_3 = \acute{z}$.

Perturbation expansion: linearizing the equations

The RANS and turbulence closure equations are highly non-linear, which makes them difficult and expensive to solve in their complete form. Solving the full RANS equations would resolve all non-linear effects. In finite-volume and finite-difference applications, the non-linear convection term is usually linearized by replacing one of the velocity variables by the velocity from the previous iteration. The cost is the very large number of iterations required for convergence on the non-linear solution.

In the present study, we postulate that linear effects are the most important for the purposes of wind resource estimation at locations where wind turbines are typically placed. In order to simplify the equations and accelerate the solution method, we disregard the non-linear effects, and choose to linearize the equations via a perturbation expansion, in which only the zero-order and first-order terms are retained, while the higher-order terms are discarded. We thus obtain two corresponding sets of equations, which are to be solved in succession:

- Zero-order equations. The zero-order flow variables are constrained to be horizontally homogeneous. These equations reduce to those describing the flow over flat terrain.
- First-order equations, whose solution then comes as a spatial correction, providing a more complete solution for flow over complex terrain.

The final solution is obtained by adding the zero- and first-order solutions together, for each of the flow variables (velocity, pressure, and turbulent quantities).

In future work, we will attempt to include some non-linear effects in the calculations, in order to improve the accuracy, especially in the lee of hills. Once the linear solution has been found, non-linear effects can be resolved by including some of the most significant second-order terms as extra source terms. These terms are calculated based on the linear solution. Inspired by Xu et al. (1994), we will direct future efforts in this direction; however the present thesis describes the development of the linear model only.

Solution method: Spectral horizontally, ODE numerical integration vertically

Motivated by our requirement for computational speed, we chose to use spectral techniques based on the Fourier transform, as they are generally less computationally expensive than finite-difference techniques. An added advantage of spectral methods is that they have a high formal order of accuracy, and tend to minimize numerical errors. They do require, however, that the domain be interpreted as periodic in the direction of Fourier transformation. This can represent a reasonable approximation in the horizontal dimensions, given a judicious sizing, buffering and periodicizing of the calculation domain (see Section 5.7).

Once Fourier-transformed in the two horizontal dimensions, the first-order equations are rearranged into a set of four (three in the 2D case) ordinary differential equations (ODEs) by algebraically eliminating the pressure and the two horizontal velocity components. The independent variable is $\acute{z}$, the transformed vertical coordinate. Since we specify boundary conditions both at the lower (ground) and upper (sky) boundaries, this constitutes a two-point boundary value problem, which we solve by numerical integration and the "shooting" technique, described in Section 5.3.

We thus have a solution method that can be called Mixed Spectral-Integration (MSI), distinct from the MSFD approach of Beljaars et al. (1987).

Storage of results in look-up tables

In the 2D case, the ODEs, when non-dimensionalized, depend on only two parameters: a non-dimensional height $k\acute{z}$, and a non-dimensional roughness length kz_0 (k is the horizontal wavenumber, z_0 is the surface roughness length). In the 3D case, an additional parameter is the angle γ between the upper-boundary forcing stress and the wavenumber vector.

This is exploited by solving the equations for all foreseeable combinations of these parameters, and storing the results into a look-up table (LUT). This is a one-time process. The flow field over any given orography is then quickly obtained by interpolating from the LUT and scaling the value of the spectral flow variables for each wavenumber component of the orography, and finally returning to real space by inverse Fourier transform.

Summary of results

RAMSIM's calculations were compared to those by various other flow models and to measurements in four test cases: two laboratory flows over idealized terrain, and two field experiments (Askervein hills and Hjardemål escarpment). We found that RAMSIM performs well over upward slopes: there is good agreement with wind speed and pressure measurements. This is also true at the top of hills or escarpments, where there is most interest in placing wind turbines, though RAMSIM like other linearized models, has a tendency to predict a spuriously large speed-up very near the ground.

The wake region is more difficult to model: the wind speed in the wake region is much overestimated by RAMSIM, and recovery occurs more rapidly than is measured. The pressure also recovers too rapidly, resulting in a reduced pressure drag. RAMSIM is not as accurate as non-linear CFD models, which are much better at capturing the wind speed reduction, pressure increase, and TKE increase in the wake region. Compared to other linear models, however, RAMSIM appears to have a slight advantage in certain lee-side wind speed calculations and in predicting the occurrence of a recirculation region (though underestimating its size). Most notably, RAMSIM is able to predict asymmetric flow over symmetric hills, while WA^{S}P gives unchanged streamlines if the flow direction is reversed.

Moreover, RAMSIM is orders of magnitude faster than non-linear CFD models, and is likely faster than even linear MSFD models: calculating the entire 2D flow field on a 2048×150 grid takes 3 seconds of CPU time.

1.8 Plan of this thesis

We now give a brief summary of how the remainder of this document is organized. In Section 2, we present the theoretical and mathematical foundation of our flow model. We state the "raw" governing equations, and describe how they will be processed to produce zero-order equations for the basic, undisturbed flow, and first-order equations for the orography-induced perturbation.

Section 3 deals with the zero-order equations for the undisturbed flow: their derivation, and, since they are comparatively simple, their solution.

Section 4 is rather technical. The first-order equations are derived and processed into their final form: four ordinary differential equations for four dependent spectral variables (vertical component of velocity and of vorticity, TKE, dissipation) with one independent variable (the transformed height $\acute{z}$). This processing is rather lengthy; the bulk of the manipulations is therefore relegated to Appendix B.

Solving the first-order equations, i.e. the ordinary differential equations derived in the previous section, represents the greatest challenge within our flow model. In Section 5 we discuss the techniques we use to do so, and analyze the associated numerical difficulties.

In Section 6, we present the results of RAMSIM calculations for the four test cases briefly mentioned above, and compare these results to measurements as well as to calculations by the numerical models of other authors.

Section 7 is devoted to the discussion and interpretation of the results presented in the previous sections, as well as the difficulties experienced and potential ways around them. In Section 8, we sum up our findings and point the way forward for the continued development of our flow model.

2 Theoretical formulation

2.1 Governing equations of the present flow model

For the remainder of this thesis, we will only be interested in mean quantities; we therefore drop the overlines for the sake of legibility (e.g. $\bar{u}_i$ is now simply designated u_i). We also ignore the molecular viscosity, such that $\nu \approx \nu_\mathrm{t}$. Furthermore, since we are interested in the mean flow over a long period of time (years), we assume steady-state flow, such that the mean properties of the flow do not change with time:

$$\frac{\partial u_i}{\partial t} = \frac{\partial E}{\partial t} = \frac{\partial \varepsilon}{\partial t} = 0$$

Our governing equations are thus the steady-state momentum transport (RANS) equation, the continuity equation, and the standard $E-\varepsilon$ closure, which consists of the TKE transport equation, and the TKE dissipation transport equation. These are listed here in order:

$$u_j \frac{\partial u_i}{\partial x_j} = -\frac{1}{\rho}\frac{\partial p}{\partial x_i} + \frac{\partial}{\partial \acute{x}_j}\left(\nu \frac{\partial u_i}{\partial x_j} + \nu \frac{\partial u_j}{\partial x_i}\right) \tag{22}$$

$$\frac{\partial u_i}{\partial x_i} = 0 \tag{23}$$

$$u_j \frac{\partial E}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{\nu}{\sigma_E}\frac{\partial E}{\partial x_j}\right) + \Pi - \varepsilon \tag{24}$$

$$u_j \frac{\partial \varepsilon}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{\nu}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x_j}\right) + \frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right) \tag{25}$$

where the production of TKE is

$$\begin{aligned} \Pi &= \nu \frac{\partial u_i}{\partial x_j}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right) \\ &= \frac{\nu}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)^2 \end{aligned} \tag{26}$$

and the (eddy) viscosity is defined as

$$\nu = C_\mu \frac{E^2}{\varepsilon} \tag{27}$$

We also define the vorticity $\boldsymbol{\omega}$ as

$$\boldsymbol{\omega} = \nabla \times \mathbf{u} \tag{28}$$

2.2 Coordinate transformation

We define the new coordinate system $\acute{x}_i$, related to the Cartesian coordinates x_i by

$$x_i = \acute{x}_i + s\lambda_i \tag{29}$$

The variable s will serve as an arbitrary expansion parameter; however, it is set to $s = 1$ in the final solution. Our aim is to design an analytical function $\lambda_j(\acute{\mathbf{x}}, \acute{z})$ that defines a terrain-following coordinate system with convenient properties. For better legibility, we will use in the following the notation $(x, y, z) = (x_1, x_2, x_3)$ and $(\acute{x}, \acute{y}, \acute{z}) = (\acute{x}_1, \acute{x}_2, \acute{x}_3)$ as well as $\acute{\mathbf{x}} = (\acute{x}_1, \acute{x}_2)$. An example of our terrain-following coordinate system is shown in Figure 5. We will now describe it in more detail.

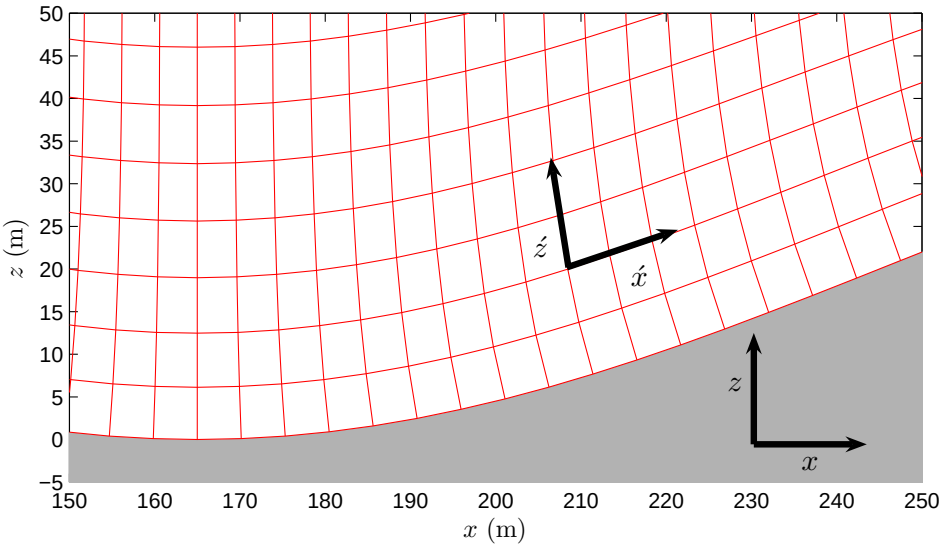


Figure 5: The terrain-following coordinate system in the two-dimensional case. The red lines represent the $(\acute{x}, \acute{z})$ coordinate lines. The grey area represents the ground.

Practical requirements for the coordinate transformation

The ground surface is defined by $z = sh_\mathrm{t}(x, y)$, where $h_\mathrm{t}(x, y)$ is the local terrain height. In order for the transformed coordinate system to be terrain-following, $\acute{z} = 0$ must correspond to this ground surface.

We want a coordinate transformation in which each Fourier component is independent of all others. We therefore formulate it as a function in the Fourier domain, $\Lambda_j(\mathbf{k}, \acute{z})$, where $\mathbf{k} = (k_1, k_2)$ is the horizontal wavenumber vector in transformed coordinates. The coordinate transformation in the real domain is then

$$\lambda_j(\acute{\mathbf{x}}, \acute{z}) = \sum_{\mathbf{k}} \Lambda_j(\mathbf{k}, \acute{z}) e^{\mathrm{i}\mathbf{k}\cdot\acute{\mathbf{x}}} \tag{30}$$

where e is the base of the natural logarithm, and the imaginary unit $\sqrt{-1}$ is written i in order to avoid confusion with the i often used as a subscript.

At the surface ($\acute{x}_3 = 0$) we would like $\lambda_1 = \lambda_2 = 0$, so that the Cartesian and transformed horizontal coordinates are identical. It would also be nice if the lines along $\acute{x}_3$ were perpendicular to the surface; this would ensure that a boundary condition in which the gradient normal to the surface of a quantity ϕ is prescribed, translates into a prescription for $\frac{\partial \phi}{\partial \acute{z}}$. Another desirable feature is $\frac{\partial \lambda_i}{\partial \acute{x}_i} = 0$, which leads to major simplifications in the equations to be written. In the far field ($\acute{x}_3 \to \infty$), we want the coordinate system to become Cartesian again: λ_i should decay with height, e.g. exponentially.

Chosen transformation

These features are attained by a transformation of the form

$$\begin{aligned}
\lambda_3(\acute{\mathbf{x}}, \acute{z}) &= \sum_{\mathbf{k}} \psi(\mathbf{k}, \acute{z}) \hat{h}_\mathrm{t}(\mathbf{k}) e^{\mathrm{i}\mathbf{k}\cdot\acute{\mathbf{x}}} \\
\lambda_j(\acute{\mathbf{x}}, \acute{z}) &= \sum_{\mathbf{k}} \frac{\mathrm{i}k_j}{k^2} \frac{\partial \psi(\mathbf{k}, \acute{z})}{\partial \acute{z}} \hat{h}_\mathrm{t}(\mathbf{k}) e^{\mathrm{i}\mathbf{k}\cdot\acute{\mathbf{x}}} \quad \text{for} \quad j = 1, 2
\end{aligned}$$

where the function $\psi(\mathbf{k}, \acute{z})$ obeys the following constraints:

$$\psi(\mathbf{k}, \acute{z}) \rightarrow 0 \quad \text{for} \quad \acute{z} \rightarrow \infty \tag{31}$$

$$\psi(\mathbf{k}, 0) = 1 \tag{32}$$

$$\left.\frac{\partial \psi(\mathbf{k}, \acute{z})}{\partial \acute{z}}\right|_{\acute{z}=0} = 0 \tag{33}$$

$$\left.\frac{\partial^2 \psi(\mathbf{k}, \acute{z})}{\partial \acute{z}^2}\right|_{\acute{z}=0} = -|\mathbf{k}|^2 \tag{34}$$

Equation (31) ensures a return to Cartesian coordinates in the far field. Equations (32) and (33) ensure that at the ground surface, $\acute{z} = 0$ and $\lambda_1 = \lambda_2 = 0$. Equation (34) ensures that $\frac{\partial \lambda_i}{\partial \acute{z}} = -\frac{\partial h_\mathrm{t}}{\partial \acute{x}_i}$, so that a zero gradient of some quantity ϕ across the ground surface translates into $\partial\phi/\partial\acute{z} = 0$. This is demonstrated as follows:

$$\begin{aligned}
\left.\frac{\partial \phi}{\partial \acute{z}}\right|_{\acute{z}=0} &= \left.\frac{\partial x_i}{\partial \acute{z}} \frac{\partial \phi}{\partial x_i}\right|_{\acute{z}=0} \\
&= \left.\left(\delta_{i3} + \frac{\partial \lambda_i}{\partial \acute{z}}\right) \frac{\partial \phi}{\partial x_i}\right|_{\acute{z}=0} \\
&= \left.\left(\frac{\partial \phi}{\partial z} - \frac{\partial h_\mathrm{t}}{\partial \acute{x}_i} \frac{\partial \phi}{\partial x_i}\right)\right|_{z=h_\mathrm{t}(x,y)} \\
&= \sqrt{1 + (\nabla h_\mathrm{t})^2}\, \mathbf{n} \cdot \nabla \phi|_{z=h_\mathrm{t}(x,y)}
\end{aligned}$$

where $\sqrt{1 + (\nabla h_\mathrm{t})^2}$ is always positive; hence $\mathbf{n} \cdot \nabla \phi = 0 \Leftrightarrow \partial\phi/\partial\acute{z} = 0$.

A suitable candidate for the role of ψ is

$$\begin{aligned}
\psi(\mathbf{k}, \acute{z}) &= (1 + |\mathbf{k}|\, \acute{z})\, e^{-|\mathbf{k}|\acute{z}} \\
\frac{\partial \psi(\mathbf{k}, \acute{z})}{\partial \acute{z}} &= -|\mathbf{k}|^2\, \acute{z} e^{-|\mathbf{k}|\acute{z}}
\end{aligned}$$

In the two-dimensional case, the transformation reduces to

$$\lambda_3(\acute{x}, \acute{z}) = \sum_k (1 + |k|\, \acute{z})\, e^{-|k|\acute{z}} \hat{h}_\mathrm{t}(k)\, e^{\mathrm{i}k\acute{x}} \tag{35}$$

$$\lambda_1(\acute{x}, \acute{z}) = \sum_k (-k\acute{z})\ e^{-|k|\acute{z}} \mathrm{i}\, \hat{h}_\mathrm{t}(k)\, e^{\mathrm{i}k\acute{x}} \tag{36}$$

In the special case of a single sinusoidal hill of wavenumber k, we have

$$\begin{aligned}
\lambda_1(\acute{x}, \acute{z}) &= 2\hat{h}_\mathrm{t}(k) \sin(k\acute{x}_1)\ |k|\acute{z}\ e^{-|k|\acute{z}} \\
\lambda_3(\acute{x}, \acute{z}) &= 2\hat{h}_\mathrm{t}(k) \cos(k\acute{x}_1)\ (1 + |k|\acute{z})\ e^{-|k|\acute{z}}
\end{aligned}$$

2.3 Transformed quantities

Scalars such as p, E, ε, and ν are invariant under coordinate transformation. Velocity, a vectorial quantity, must undergo a transformation. The relationship between the transformed velocity $\acute{u}_i$ and the velocity u_j can be uncovered by considering the substantive derivative operator (a.k.a. convective or Lagrangian or material derivative), which is valid for all coordinate systems:

$$\begin{aligned}
\frac{D}{Dt} &= \frac{\partial}{\partial t} + \mathbf{u} \cdot \nabla \\
&= \frac{\partial}{\partial t} + \acute{\mathbf{u}} \cdot \acute{\nabla}
\end{aligned}$$

Applying this operator in both forms to x_i, we get

$$\frac{\partial x_i}{\partial t} + u_j \frac{\partial x_i}{\partial x_j} = \frac{\partial x_i}{\partial t} + \acute{u}^j \frac{\partial x_i}{\partial \acute{x}_j}$$

Since the coordinate x_i is independent of time, and since $\partial x_j / \partial x_i = \delta_{ij}$, we are left with

$$\acute{u}^j = \frac{\partial \acute{x}_j}{\partial x_i} u_i$$

However, we will work with the convenient variable $\acute{U}_j = J\acute{u}^j$, which, as we will see, will allow us to write the continuity equation in its familiar, concise form despite being in transformed coordinates. Indeed, given a volume element dV that maps to $d\acute{V}$ in the transformed coordinate system, the momentum dp_i contained in dV is $dp_i = \rho u_i dV$ and is related to the transformed momentum $d\acute{p}_j$ in $d\acute{V}$ by

$$d\acute{p}_j = \frac{\partial \acute{x}_j}{\partial x_i} dp_i = \frac{\partial \acute{x}_j}{\partial x_i} \rho u_i dV = \frac{\partial \acute{x}_j}{\partial x_i} \rho \frac{\partial x_i}{\partial \acute{x}_j} \acute{u}_j J d\acute{V} = \rho \acute{U}_j d\acute{V}$$

Hence the use of $\acute{U}_j$ preserves the expression for the momentum when switching to the transformed coordinates.

2.4 The equations in transformed coordinates

We now reformulate the governing equations (22) to (26) with all quantities expressed in transformed coordinates. We start by noting that one can demonstrate the useful relation

$$J \frac{\partial \phi}{\partial x_j} = J \frac{\partial \acute{x}_q}{\partial x_j} \frac{\partial \phi}{\partial \acute{x}_q} = \frac{\partial}{\partial \acute{x}_q} \left(J \frac{\partial \acute{x}_q}{\partial x_j} \phi \right) \tag{37}$$

where ϕ is any arbitrary quantity. This will be used in the derivations below.

Continuity equation

We reformulate the continuity equation in terms of $\acute{U}_j$:

$$\frac{\partial u_i}{\partial x_i} = \frac{\partial \acute{x}_q}{\partial x_i} \frac{\partial}{\partial \acute{x}_q} \left(\frac{\partial x_i}{\partial \acute{x}_j} \frac{\acute{U}_j}{J} \right) = 0$$

Expanding the derivative of the product,

$$\frac{\partial \acute{x}_q}{\partial x_i} \left(\acute{U}_j \frac{\partial}{\partial \acute{x}_q} \left(\frac{1}{J} \frac{\partial x_i}{\partial \acute{x}_j} \right) + \frac{1}{J} \frac{\partial x_i}{\partial \acute{x}_j} \frac{\partial}{\partial \acute{x}_q} \left(\acute{U}_j \right) \right)$$

$$= \acute{U}_j \frac{\partial}{\partial x_i} \left(\frac{1}{J} \frac{\partial x_i}{\partial \acute{x}_j} \right) + \frac{1}{J} \frac{\partial}{\partial \acute{x}_j} \acute{U}_j = 0$$

It can be shown that the first term is zero, by expanding its derivative of a product and by noting the identity

$$\delta J = \delta(\det \mathbf{J}) = \det \mathbf{J} \ \mathrm{Tr}(\mathbf{J}^{-1} \delta \mathbf{J})$$

hence

$$\frac{\partial J}{\partial x_i} = J \frac{\partial \acute{x}_q}{\partial x_p} \frac{\partial}{\partial x_i} \frac{\partial x_p}{\partial \acute{x}_q}$$

So the continuity equation can be expressed in the same form as in the original equation (23):

$$\frac{\partial \acute{U}_j}{\partial \acute{x}_j} = 0 \tag{38}$$

TKE equation

Starting with the original equation (24),

$$u_j \frac{\partial E}{\partial x_j} = \frac{\partial}{\partial x_j}\left(\frac{\nu}{\sigma_E}\frac{\partial E}{\partial x_j}\right) + \Pi - \varepsilon$$

and using the chain rule, we get

$$\frac{1}{J}\frac{\partial x_j}{\partial \acute{x}_i}\acute{U}_i \frac{\partial \acute{x}_i}{\partial x_j}\frac{\partial E}{\partial \acute{x}_i} = \frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_E}\frac{\partial \acute{x}_q}{\partial x_j}\frac{\partial E}{\partial \acute{x}_q}\right) + \Pi - \varepsilon$$

which can be rearranged as

$$\acute{U}_i \frac{\partial E}{\partial \acute{x}_i} = \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_E} J \frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial \acute{x}_q}{\partial x_j}\frac{\partial E}{\partial \acute{x}_q}\right) + J(\Pi - \varepsilon)$$

and, using the relation in Equation (37),

$$\acute{U}_i \frac{\partial E}{\partial \acute{x}_i} = \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_E}\frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial}{\partial \acute{x}_q}\Big(J\frac{\partial \acute{x}_q}{\partial x_j}E\Big)\right) + J(\Pi - \varepsilon) \tag{39}$$

Dissipation equation

We follow the same steps as for the TKE equation:

$$\begin{aligned}
u_j \frac{\partial \varepsilon}{\partial x_j} &= \frac{\partial}{\partial x_j}\left(\frac{\nu}{\sigma_\varepsilon}\frac{\partial \varepsilon}{\partial x_j}\right) + \frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right) \\
\frac{1}{J}\frac{\partial x_j}{\partial \acute{x}_i}\acute{U}_i \frac{\partial \acute{x}_i}{\partial x_j}\frac{\partial \varepsilon}{\partial \acute{x}_i} &= \frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_\varepsilon}\frac{\partial \acute{x}_q}{\partial x_j}\frac{\partial \varepsilon}{\partial \acute{x}_q}\right) + \frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right) \\
\acute{U}_i \frac{\partial \varepsilon}{\partial \acute{x}_i} &= \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_\varepsilon} J \frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial \acute{x}_q}{\partial x_j}\frac{\partial \varepsilon}{\partial \acute{x}_q}\right) + J\frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right) \\
\acute{U}_i \frac{\partial \varepsilon}{\partial \acute{x}_i} &= \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_\varepsilon}\frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial}{\partial \acute{x}_q}\Big(J\frac{\partial \acute{x}_q}{\partial x_j}\varepsilon\Big)\right) \\
&\quad + J\frac{\varepsilon}{E}\left(C_{\varepsilon 1}\Pi - C_{\varepsilon 2}\varepsilon\right)
\end{aligned} \tag{40}$$

Production of TKE

We start with the original equation (26) in Cartesian coordinates,

$$\Pi = \frac{\nu}{2}\left(\frac{\partial u_i}{\partial x_j} + \frac{\partial u_j}{\partial x_i}\right)^2$$

By using the chain rule and the relation in Equation (37), we get:

$$\begin{aligned}
\Pi &= \frac{\nu}{2}\left(\frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial}{\partial \acute{x}_p}\Big(\frac{1}{J}\frac{\partial x_i}{\partial \acute{x}_r}\acute{U}_r\Big) + \frac{\partial \acute{x}_p}{\partial x_i}\frac{\partial}{\partial \acute{x}_p}\Big(\frac{1}{J}\frac{\partial x_j}{\partial \acute{x}_r}\acute{U}_r\Big)\right)^2 \\
&= \frac{\nu}{2J^2}\left(\frac{\partial}{\partial \acute{x}_p}\Big(\frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial x_i}{\partial \acute{x}_r}\acute{U}_r\Big) + \frac{\partial}{\partial \acute{x}_p}\Big(\frac{\partial \acute{x}_p}{\partial x_i}\frac{\partial x_j}{\partial \acute{x}_r}\acute{U}_r\Big)\right)^2 \\
&= \frac{\nu}{2J^2}\left[\frac{\partial}{\partial \acute{x}_p}\left(\Big(\frac{\partial \acute{x}_p}{\partial x_j}\frac{\partial x_i}{\partial \acute{x}_r} + \frac{\partial \acute{x}_p}{\partial x_i}\frac{\partial x_j}{\partial \acute{x}_r}\Big)\acute{U}_r\right)\right]^2
\end{aligned} \tag{41}$$

Momentum transport equation

Similar derivations leads to the following expression for the momentum transport equation in transformed coordinates:

$$\frac{\partial}{\partial \acute{x}_q}\left(\frac{\partial \acute{x}_q}{\partial x_j}\nu\frac{\partial}{\partial \acute{x}_r}\left(\Big(\frac{\partial \acute{x}_r}{\partial x_j}\frac{\partial x_i}{\partial \acute{x}_p} + \frac{\partial \acute{x}_r}{\partial x_i}\frac{\partial x_j}{\partial \acute{x}_p}\Big)\acute{U}_p\right)\right) - \frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J\frac{\partial \acute{x}_q}{\partial x_i}p\right) = \acute{U}_j\frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J}\frac{\partial x_i}{\partial \acute{x}_q}\acute{U}_q\right) \tag{42}$$

2.5 Perturbation expansion

Generalities

Any quantity f can be expressed as a function of an arbitrary expansion parameter, s, and, barring singularities, expanded about $s = 0$ as

$$f(s) \quad = \quad f(0) + s \left.\frac{\partial f(s)}{\partial s}\right|_{s=0} + \frac{s^2}{2} \left.\frac{\partial^2 f(s)}{\partial s^2}\right|_{s=0} + \dots$$

This Taylor series expansion can be written more compactly as

$$f(s) \quad = \quad f_0 + s f_1 + s^2 f_2 + \dots$$

where

$$\begin{aligned} f_0 &= f(s)|_{s=0} \\ f_1 &= \left.\frac{\partial f(s)}{\partial s}\right|_{s=0} \\ &\text{etc.} \end{aligned}$$

One can choose to keep only the zero-order and first-order terms of the expansion, and neglect the rest. In this case, the expansion is good for "small" s, or rather

$$\begin{aligned} s f_1 &< f_0 \\ s^2 f_2 + ... &\ll f_0 + s f_1 \end{aligned}$$

Application

In our model, we choose to have the s parameterize the coordinate transformation as $\acute{x}_j = x_j - s\lambda_j$. Intuitively put, s represents the degree of departure from flat terrain toward real terrain, i.e. at $s = 0$, the terrain is flat and the coordinates remain plainly Cartesian. The zero-order quantities, f_0, represent the quantities calculated when $s = 0$, i.e. over flat terrain. They are therefore horizontally homogeneous. As s increases, the terrain is "pumped up" and the coordinate system approaches its fully transformed state. At $s = 1$, the terrain is fully realistic (at the lower boundary, $z = h_t(x, y)$) and the coordinate transformation is full ($\acute{x}_j = x_j - \lambda_j$). The proper equations (for the desired terrain) are obtained when we set $s = 1$. The first-order (linear) approximation's validity thus requires

$$f_2 + f_3 + ... \quad \ll \quad f_0 + f_1 \tag{43}$$

This requirement need not be strictly satisfied at every point in the domain, but rather only in the general sense that, all Fourier components taken into account, higher-order effects have less impact on the flow than zero- and first-order effects, especially at points of interest.

Example of a perturbed quantity

What is the expression for $(E^{\frac{3}{2}})_1$? It is simply

$$\begin{aligned}
(E^{\frac{3}{2}})_1 &= \left.\frac{\partial(E^{\frac{3}{2}})}{\partial s}\right|_{s=0} \\
&= \left.\left(\frac{3}{2}E^{\frac{1}{2}}\frac{\partial E}{\partial s}\right)\right|_{s=0} \\
&= \left.\left(\frac{3}{2}\left(E_0+sE_1+...\right)^{\frac{1}{2}}\frac{\partial\left(E_0+sE_1+s^2E_2+...\right)}{\partial s}\right)\right|_{s=0} \\
&= \left.\left(\frac{3}{2}\left(E_0+sE_1+...\right)^{\frac{1}{2}}\left(E_1+2\,sE_2+...\right)\right)\right|_{s=0} \\
&= \frac{3}{2}\left(E_0\right)^{\frac{1}{2}}E_1
\end{aligned}$$

Other perturbed quantities are derived in a similar fashion.

Perturbed metrics

We define the metrics

$$\begin{aligned}
\beta_j^i &= \frac{\partial x_i}{\partial \acute{x}_j} \\
\alpha_i^j &= \frac{\partial \acute{x}_j}{\partial x_i} = \left(\frac{\partial x_i}{\partial \acute{x}_j}\right)^{-1} \\
J &= \det\mathbf{J} = \det\beta_j^i = \det\frac{\partial x_i}{\partial \acute{x}_j}
\end{aligned}$$

We insert the coordinate transformation stated in Equation (29) into the expressions for these metrics.

$$\beta_j^i = \frac{\partial(\acute{x}_i+s\lambda_i)}{\partial \acute{x}_j} = \delta_{ij}+s\frac{\partial\lambda_i}{\partial \acute{x}_j}$$

Thus $\beta^i_{j(0)}=\delta_{ij}$ and $\beta^i_{j(1)}=\frac{\partial\lambda_i}{\partial \acute{x}_j}$.

$$\alpha_i^j = \left(\frac{\partial(\acute{x}_i+s\lambda_i)}{\partial \acute{x}_j}\right)^{-1} = ... = \delta_{ij}-s\frac{\partial\lambda_i}{\partial \acute{x}_j}+\mathcal{O}\left(s^2\right)$$

where we used the relation $d\mathbf{A}^{-1}=-\mathbf{A}^{-1}d\mathbf{A}\mathbf{A}^{-1}$, and where $\mathcal{O}\left(\star\right)$ indicates the order of the leading term of the remainder of the series. Thus $\alpha^i_{j(0)}=\delta_{ij}$ and $\alpha^i_{j(1)}=-\frac{\partial\lambda_i}{\partial \acute{x}_j}$.

$$\begin{aligned}
J &= \det\frac{\partial(\acute{x}_i+s\lambda_i)}{\partial \acute{x}_j} = \det\left(\delta_{ij}+s\frac{\partial\lambda_i}{\partial \acute{x}_j}\right) \\
&= \det\begin{pmatrix} 1+s\frac{\partial\lambda_1}{\partial \acute{x}_1} & s\frac{\partial\lambda_1}{\partial \acute{x}_2} & s\frac{\partial\lambda_1}{\partial \acute{x}_3} \\ s\frac{\partial\lambda_2}{\partial \acute{x}_1} & 1+s\frac{\partial\lambda_2}{\partial \acute{x}_2} & s\frac{\partial\lambda_2}{\partial \acute{x}_3} \\ s\frac{\partial\lambda_3}{\partial \acute{x}_1} & s\frac{\partial\lambda_3}{\partial \acute{x}_2} & 1+s\frac{\partial\lambda_3}{\partial \acute{x}_3} \end{pmatrix} \\
&= 1+s\frac{\partial\lambda_i}{\partial \acute{x}_i}+\mathcal{O}\left(s^2\right)
\end{aligned}$$

Thus $J_0=1$ and $J_1=\frac{\partial\lambda_i}{\partial \acute{x}_i}$.

2.6 Fourier transformation

Once the first-order equations are expressed in the terrain-following transformed coordinate system, they are Fourier-transformed along the two horizontal dimensions. We now briefly review the concepts and relations used in this process.

Definition of symbols and concepts

The two-dimensional horizontal Fourier transform of a field variable $\phi(\acute{x}_1, \acute{x}_2, \acute{x}_3)$ is defined as:

$$\hat{\phi}(k_1, k_2, \acute{x}_3) = \frac{1}{2\pi} \int \int_{-\infty}^{\infty} \phi(\acute{x}_1, \acute{x}_2, \acute{x}_3) e^{-\mathrm{i}(k_1 \acute{x}_1 + k_2 \acute{x}_2)} d\acute{x}_1 d\acute{x}_2$$

where k_1 and k_2 are the wave numbers and $\hat{\phi}$ denotes the Fourier-transformed variable. The corresponding inverse Fourier transform is:

$$\phi(\acute{x}_1, \acute{x}_2, \acute{x}_3) = \frac{1}{2\pi} \int \int_{-\infty}^{\infty} \hat{\phi}(k_1, k_2, \acute{x}_3) e^{\mathrm{i}(k_1 \acute{x}_1 + k_2 \acute{x}_2)} dk_1 dk_2$$

The horizontal discrete Fourier transform (DFT), as its name implies, is a discrete version of the Fourier transform, applied when a quantity ϕ is defined by a finite number of samples in a domain of finite size $L_1 \times L_2$. Given N_1 samples at regular intervals $\Delta\acute{x}_1 = L_1/N_1$ along $\acute{x}_1$ and N_2 samples at regular intervals $\Delta\acute{x}_2 = L_2/N_2$ along $\acute{x}_2$, the set of $N_1 \times N_2$ complex numbers ϕ_{q_1,q_2} (where $q_1 = 0, \ldots, N_1 - 1$ and $q_2 = 0, \ldots, N_2 - 1$) is transformed into the sequence of $N_1 \times N_2$ complex numbers $\hat{\phi}_{m_1,m_2}$ (where $m_1 = 0, \ldots, N_1 - 1$ and $m_2 = 0, \ldots, N_2 - 1$) as

$$\hat{\phi}_{m_1,m_2} = \sum_{q_1=0}^{N_1-1} \sum_{q_2=0}^{N_2-1} \phi_{q_1,q_2} e^{-2\pi\mathrm{i}(\frac{m_1 q_1}{N_1} + \frac{m_2 q_2}{N_2})}$$

The corresponding inverse discrete Fourier transform (IDFT) is given by

$$\phi_{q_1,q_2} = \frac{1}{N_1 N_2} \sum_{m=0}^{N-1} \hat{\phi}_{m_1,m_2} e^{2\pi\mathrm{i}(\frac{m_1 q_1}{N_1} + \frac{m_2 q_2}{N_2})}$$

Noting that the wavenumber vector is $\mathbf{k} = (k_1, k_2) = (2\pi m_1/N_1\Delta\acute{x}_1, 2\pi m_2/N_2\Delta\acute{x}_1)$ and the horizontal position vector is $\acute{\mathbf{x}} = (\acute{x}_1, \acute{x}_2)$ the DFT and IDFT can also be written more succinctly as, respectively,

$$\hat{\phi}(\mathbf{k}, \acute{x}_3) = \sum_{\text{all } \acute{\mathbf{x}}} \phi(\acute{\mathbf{x}}, \acute{x}_3) e^{-\mathrm{i}\mathbf{k}\cdot\acute{\mathbf{x}}}$$

and

$$\phi(\acute{\mathbf{x}}, \acute{x}_3) = \frac{1}{N_1 N_2} \sum_{\text{all } \mathbf{k}} \hat{\phi}(\mathbf{k}, \acute{x}_3) e^{\mathrm{i}\mathbf{k}\cdot\acute{\mathbf{x}}}$$

In practice, we compute the DFT and IDFT using a fast Fourier transform (FFT) algorithm called FFTW, developed by Frigo and Johnson (1998, 2005). In the remainder of this document, we will indicate the forward and inverse Fourier transforms as $\hat{\phi} = \mathcal{F}\{\phi\}$ and $\phi = \mathcal{F}^{-1}\left\{\hat{\phi}\right\}$.

Spectral representation of equations

Equations in real space have a corresponding spectral representation in Fourier space. For example, consider the equation in real space

$$f_0\, a + \frac{\partial b}{\partial \acute{x}_1} + \frac{\partial c}{\partial \acute{x}_3} = 0 \tag{44}$$

where f_0 is horizontally homogeneous and $a(\acute{x}_1, \acute{x}_2, \acute{x}_3)$, $b(\acute{x}_1, \acute{x}_2, \acute{x}_3)$ and $c(\acute{x}_1, \acute{x}_2, \acute{x}_3)$ are three field variables. The corresponding spectral representation is derived from the following relationships, where the field arguments are dropped for notational convenience:

$$\begin{aligned}
f_0\, a &= f_0\, \mathcal{F}^{-1}\{\hat{a}\} = \mathcal{F}^{-1}\{f_0\, \hat{a}\} \\
\frac{\partial b}{\partial \acute{x}_1} &= \mathcal{F}^{-1}\left\{ \mathrm{i} k_1 \hat{b} \right\} \\
\frac{\partial c}{\partial \acute{x}_3} &= \frac{\partial}{\partial \acute{x}_3} \mathcal{F}^{-1}\{\hat{c}\} = \mathcal{F}^{-1}\left\{ \frac{\partial \hat{c}}{\partial \acute{x}_3} \right\}
\end{aligned}$$

and

$$\mathcal{F}^{-1}\{X\} + \mathcal{F}^{-1}\{Y\} = \mathcal{F}^{-1}\{X + Y\}$$

Hence

$$\mathcal{F}^{-1}\left\{ f_0\, \hat{a} + \mathrm{i} k_1 \hat{b} + \frac{\partial \hat{c}}{\partial \acute{x}_3} \right\} = 0$$

or simply

$$f_0\, \hat{a} + \mathrm{i} k_1 \hat{b} + \frac{\partial \hat{c}}{\partial \acute{x}_3} = 0$$

which is the Fourier-transformed version of Equation (44). Note that horizontally homogeneous quantities, such as our zero-order quantities, are treated as constants by the horizontal Fourier transformation.

2.7 Boundary conditions

Our calculation domain has a finite size, delimited by upper, lower, and lateral boundaries. We must specify what happens to our field variables at the edges of this domain. The lateral boundaries represent the edges of the topographical map above which we are to calculate the flow field. Our spectral treatment of the equations require us to consider all quantities to be horizontally periodic, and so we have periodic lateral boundary conditions. The lower boundary and upper boundaries respectively represent the ground and "high up in the sky", a place where the orography-induced perturbations vanish. We will now express our lower and upper boundary conditions in more mathematical terms.

Lower boundary

The lower boundary, $\acute{z} = 0$, is a rough wall placed at a height z_0 above the actual ground surface, where z_0 is the surface roughness length. We assume that the flow near a rough wall resembles a constant-flux layer. Let $\mathbf{n}$ be the unit vector normal to the ground surface. The lower boundary conditions are:

$$\mathbf{u} = 0 \tag{45}$$

$$\mathbf{n} \cdot \nabla(\mathbf{n} \cdot \mathbf{u}) = 0 \tag{46}$$

$$\mathbf{n} \cdot \nabla E = 0 \tag{47}$$

$$\varepsilon = \frac{C_\mu^{\frac{3}{4}} E^{\frac{3}{2}}}{\kappa z_0} \tag{48}$$

and in the 3D case,

$$\mathbf{n} \cdot \boldsymbol{\omega} = 0 \tag{49}$$

where $\boldsymbol{\omega} \equiv \nabla \times \mathbf{u}$ is the vorticity. Equation (49) can also be written as $\acute{\omega}_3 = 0$, where $\acute{\omega}_3$ is the vertical component of the vorticity. Equation (45) indicates a no-slip boundary condition. Equation (46) translates into $\partial \acute{u}_3 / \partial \acute{z} = 0$, and follows from the continuity equation:

$$\begin{aligned} 0 \equiv \nabla \cdot \mathbf{u} &= \nabla_i (n_i n_j u_j + u_i - n_i n_j u_j) \\ &= \mathbf{n} \cdot \nabla (\mathbf{n} \cdot \mathbf{u}) + \underbrace{\nabla_i (\delta_{ij} - n_i u_i) u_j}_{=0} \end{aligned}$$

where the last term is zero because it represents surface-following differentiation, yet u_j is uniformly zero on the surface. Similarly, Equation (49) can be written as

$$\mathbf{n} \cdot (\nabla \times \mathbf{u}) = (\mathbf{n} \times \nabla) \cdot \mathbf{u} = 0$$

where $\mathbf{n} \times \nabla$ represents surface-following differentiation, necessarily zero. Equations (47) and (48) follow from our assumption of a constant-stress layer, which we will explain in Section 3.2.

Upper boundary

The upper boundary, $\acute{z} = \acute{z}_{\text{top}}$, is located high enough for $e^{-|\mathbf{k}|\acute{z}_{\text{top}}}$ to be small, i.e. where the coordinates become Cartesian again and all perturbations become negligible, but not so high as to cause numerical instability problems (discussed in Section 5.5, page 69). We found $|\mathbf{k}|\acute{z}_{\text{top}} = 10$ to be satisfactory ($e^{-10} = 4.5 \times 10^{-5}$). Of course, fixing the boundary at $|\mathbf{k}|\acute{z}_{\text{top}} = 10$ means that the calculation domain ends at a different height $\acute{z}_{\text{top}}$ for each wavenumber component $\mathbf{k}$, as illustrated schematically in Figure 6. We assume that the contribution of a given wavenumber

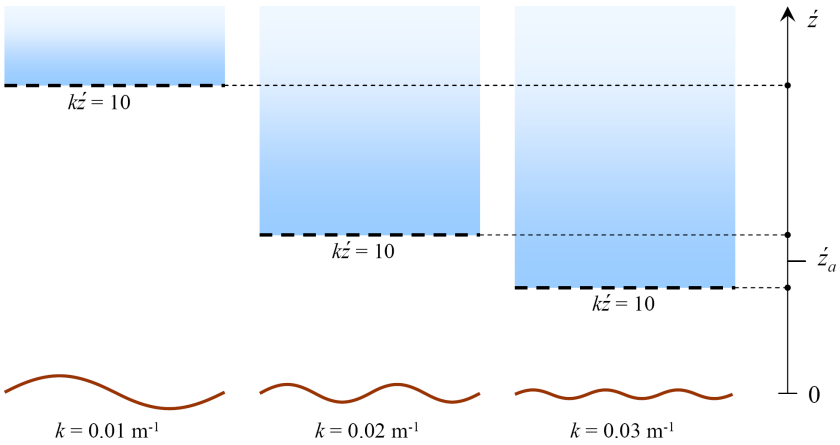


Figure 6: Location of the upper boundary for three wavenumber components of a hypothetical terrain. Above the upper boundary for a given wavenumber (blue area), the perturbation is assumed to be zero. Diagram is not to scale.

to the perturbation at some height of interest $\acute{z}_a$ is zero if $k\acute{z}_a > 10$, i.e. if $\acute{z}_a$ is above the upper boundary for this particular wavenumber (e.g. $k = 0.03$ m^{-1} on Figure 6). In our calculations, the perturbation does indeed fall to negligible levels as the upper boundary is approached.

We therefore make the assumption that the upper boundary is high enough to be a reasonable approximation to $z \to \infty$, where the flow becomes purely horizontal.

The upper boundary-conditions are then:

$$\acute{u}_3 = 0 \tag{50}$$

$$\Pi = \varepsilon \tag{51}$$

$$\frac{\partial E}{\partial x_3} = 0 \tag{52}$$

$$\tau_{3q} = \text{constant} \quad \text{for} \quad q = 1, 2 \tag{53}$$

and in the 3D case,

$$\acute{\omega}_3 = 0 \tag{54}$$

Equations (50), (51) and (52) represent the return to purely horizontal flow, equilibrium turbulence, and uniform TKE, respectively, as the orography-induced flow perturbation dies out with height. Equation (54) follows from the same arguments as the no-vorticity lower boundary condition in Equation (49). Imposing a constant forcing stress τ_{3q} (downward flux of horizontal momentum) at the top can be spelled out as

$$\begin{aligned} \tau_{3q} &= \nu \left(\frac{\partial u_q}{\partial x_3} + \frac{\partial u_3}{\partial x_q} \right) \\ &= C_\mu \frac{E^2}{\varepsilon} \frac{\partial u_q}{\partial x_3} \end{aligned} \tag{55}$$

where we used $u_3 \to 0$. In Equation (64) (Section 3.2), we will define the friction velocity $U_* \equiv (-\overline{u_1' u_3'})^{\frac{1}{2}} = \sqrt{-\tau_{13}}$, which we will use as a convenient scaling velocity. Hence in the two-dimensional case we can write

$$C_\mu \frac{E^2}{\varepsilon} \frac{\partial u}{\partial x_3} = -U_*^2 \tag{56}$$

where u is the horizontal velocity. The constant U_* can be used for proper scaling, i.e. chosen such that the velocity at a certain point in the domain conforms to some measurement. In the three-dimensional case, the forcing stress has two horizontal components, and so we have to specify the angle γ between the direction of the forcing stress and the wavenumber vector $\mathbf{k}$.

3 The zero-order equations: derivation and solution

As stated previously, the zero-order flow variables $\acute{U}_i^{(0)}$, p_0, E_0, ε_0, and ν_0 are all horizontally homogeneous; hence the zero-order equations are one-dimensional in $\acute{x}_3$. The equations under these conditions are well known. For the sake of completeness, we nevertheless summarize here their derivation from the full governing equations; the zero-order equations are readily obtained as many of their terms vanish due to horizontal homogeneity. The solution to the flow equations under conditions of horizontal homogeneity has also been known for decades (see for instance Monin and Yaglom 1971 or Businger 1973). We will summarize its derivation here and introduce the concept of surface layer.

3.1 Deriving the zero-order equations

Continuity equation

The zero-order continuity equation is

$$\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_j} = \frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_1} + \frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_2} + \frac{\partial \acute{U}_3^{(0)}}{\partial \acute{x}_3} = 0$$

Since $\acute{U}_j^{(0)}$ is horizontally homogeneous, we are left with

$$\frac{\partial \acute{U}_3^{(0)}}{\partial \acute{x}_3} = 0$$

i.e. $\acute{U}_3^{(0)}$ is uniform throughout the domain. Since there is no flow ($\acute{U}_3^{(0)} = 0$) across the lower boundary, we obtain the simple result

$$\acute{U}_3^{(0)} = 0 \quad \text{everywhere.} \tag{57}$$

Momentum transport equation

Using only zero-order quantities, we can write the zero-order momentum transport equation as

$$\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) - \frac{\partial p_0}{\partial \acute{x}_i} = \acute{U}_j^{(0)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}$$

Examining the advection term,

$$\acute{U}_j^{(0)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} = \acute{U}_1^{(0)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_1} + \acute{U}_2^{(0)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_2} + \acute{U}_3^{(0)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3} = 0$$

we observe that it vanishes, since $\acute{U}_i^{(0)}$ is horizontally homogeneous and $\acute{U}_3^{(0)} = 0$. For the same reason, the off-diagonal diffusion term vanishes:

$$\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right) = \frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_i}\right) + \frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_i}\right) + \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial \acute{U}_3^{(0)}}{\partial \acute{x}_i}\right) = 0$$

For the cases $i = 1, 2$, the pressure term vanishes since p_0 is horizontally homogeneous. We can thus write the zero-order momentum transport equations as

$$\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_3}\right) = 0 \qquad \text{and} \qquad \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_3}\right) = 0 \tag{58}$$

For the case $i = 3$, the diffusion and advection terms vanish, since $\acute{U}_3^{(0)} = 0$. The vertical momentum transport equation thus reduces to

$$\frac{\partial p_0}{\partial \acute{x}_3} = 0$$

i.e. p_0 is uniform throughout the domain. Since the mean value of pressure has no significance (only its gradient does), we can arbitrarily choose it to be $p_0 = 0$.

TKE equation

$$\acute{U}_j^{(0)} \frac{\partial E_0}{\partial \acute{x}_j} = \frac{\partial}{\partial \acute{x}_j} \left(\frac{\nu_0}{\sigma_E} \frac{\partial E_0}{\partial \acute{x}_j} \right) + \Pi_0 - \varepsilon_0$$

As in the momentum equation, the advection term vanishes. Due to horizontal homogeneity and $\acute{U}_3^{(0)} = 0$, the production term reduces to

$$\begin{aligned}
\Pi_0 &= \frac{\nu_0}{2} \left(\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_q} \right)^2 \\
&= \frac{\nu_0}{2} \left(\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_q} \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_q} + 2 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_q} \right) \\
&= \frac{\nu_0}{2} \left(2 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} + 2 \underbrace{\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_j} \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_q}}_{=0} \right) \\
&= \nu_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3} \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}
\end{aligned} \tag{59}$$

for $q = 1, 2$. Hence the zero-order TKE transport equation is:

$$\frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \frac{\partial E_0}{\partial \acute{x}_3} \right) + \Pi_0 - \varepsilon_0 = 0 \tag{60}$$

Dissipation equation

$$\acute{U}_i^{(0)} \frac{\partial \varepsilon_0}{\partial \acute{x}_i} = \frac{\partial}{\partial \acute{x}_j} \left(\frac{\nu_0}{\sigma_\varepsilon} \frac{\partial \varepsilon_0}{\partial \acute{x}_j} \right) + \frac{\varepsilon_0}{E_0} \left(C_{\varepsilon 1} \Pi_0 - C_{\varepsilon 2} \varepsilon_0 \right)$$

As before, the advection term vanishes, and the production Π_0 reduces as in Equation (59). The zero-order dissipation equation is therefore

$$\frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_\varepsilon} \frac{\partial \varepsilon_0}{\partial \acute{x}_3} \right) + \frac{\varepsilon_0}{E_0} \left(C_{\varepsilon 1} \Pi_0 - C_{\varepsilon 2} \varepsilon_0 \right) = 0 \tag{61}$$

3.2 The surface layer

The RANS equations (8) for horizontally homogeneous flow can be written as

$$\frac{\partial u_i}{\partial t} + C_i = -\frac{\partial \overline{u_i' u_3'}}{\partial x_3} \tag{62}$$

where $i = 1, 2$ and where C_i represents the Coriolis force, which we re-include to generalize this discussion. Representing the LHS of this equation by $\mathcal{H}_i$ and integrating from the ground up yields

$$\int_0^z \mathcal{H}_i = \overline{u_i' u_3'}\Big|_0 - \overline{u_i' u_3'}\Big|_z \tag{63}$$

We can choose the upper limit of integration to be a certain height $z = h_s$, at which the integral of $\mathcal{H}_i$ is small, e.g. 10% of $\overline{u_i' u_3'}\Big|_0$. Doing so defines a *surface layer* of height h_s within which the Reynolds stresses are approximately constant: $\overline{u_i' u_3'}\Big|_z \approx \overline{u_i' u_3'}\Big|_0$. In the atmosphere, h_s is typically of the order of 1 to 100 m, or about 10% of the ABL. In our case, since we neglect the Coriolis force and assume a steady state ($\partial u_i/\partial t = 0$), Equation (62) straightforwardly implies a constant stress.

By choosing a coordinate system in which x_1 is aligned with the wind flow near the surface, we can define the convenient scaling quantity,

$$U_* = \left(-\overline{u_1' u_3'}\right)^{\frac{1}{2}} \tag{64}$$

U_* is called the friction velocity, and is taken to be constant with z throughout the stationary, horizontally homogeneous and neutral surface layer.

3.3 Homogeneous flow in the surface layer: solution

Let us restate the zero-order equations:

$$\begin{aligned}
\acute{U}_3^{(0)} &= 0 \\
\frac{\partial}{\partial \acute{z}}\left(\nu_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{z}}\right) &= 0 \\
p_0 &= 0 \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) + \Pi_0 - \varepsilon_0 &= 0 \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) + \frac{\varepsilon_0}{E_0}\left(C_{\varepsilon 1}\Pi_0 - C_{\varepsilon 2}\varepsilon_0\right) &= 0 \\
\Pi_0 &= \nu_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}
\end{aligned}$$

where $q = 1, 2$. These are in effect the equations for horizontally homogeneous flow in the surface layer over flat terrain. An analytical solution to these equations has been known for decades; it can be obtained semi-empirically, provided certain assumptions are made. These assumptions, though requiring a bit of hand-waving, have been confirmed as valid by a considerable body of experimental evidence. We summarize here the derivation of this solution; Monin and Yaglom (1971) present a more rigourous and complete derivation.

The flux of the momentum imparted to the surface by turbulent transfer can be written as

$$F_m = -\rho \nu_t \frac{\partial u}{\partial \acute{z}} \tag{65}$$

where ν_t is the eddy viscosity. By definition,

$$F_m = -\rho \overline{u'w'} = \tau_w \tag{66}$$

where the surface stress, τ_w, is the drag force per unit area on the surface exerted by the momentum transfer down to the surface. In Equation (64), we defined the friction velocity, which we now use as a convenient scaling velocity, assuming a constant-stress surface layer: $U_* \equiv (\tau_w/\rho)^{\frac{1}{2}} = \left(-\overline{u'w'}\right)^{\frac{1}{2}}$. We can thus re-write Equation (65) as

$$\nu_t \frac{\partial u}{\partial \acute{z}} = U_*^2 \tag{67}$$

One must now make an assumption about ν_t; from dimensional arguments one can write

$$\nu_t = \kappa U_* \acute{z} \tag{68}$$

where κ is von Kármán's constant, a proportionality constant evaluated empirically to be equal to 0.4. Substituting this into Equation (67) yields

$$\frac{\partial u}{\partial \acute{z}} = \frac{U_*}{\kappa \acute{z}}$$

This expression is impractical, since it is singular at $\acute{z} = 0$, leading to infinite shear at the surface. We therefore introduce a surface roughness length, z_0, defined such that $U = 0$ at $\acute{z} = 0$:

$$\frac{\partial u}{\partial \acute{z}} = \frac{U_*}{\kappa\,(\acute{z} + z_0)} \tag{69}$$

Integrating this expression yields the well-known logarithmic wind profile:

$$u = \frac{U_*}{\kappa} \ln\left(\frac{\acute{z} + z_0}{z_0}\right) \tag{70}$$

It is readily shown that under neutral conditions, and assuming we are in a surface layer where the momentum flux (a.k.a. shear stress) is uniform from the top of the layer to the ground, i.e. uniform TKE, the TKE transport equation for a horizontally homogeneous flow reduces to an equilibrium between the production and dissipation of TKE:

$$\begin{aligned} \varepsilon &= \Pi \\ &= U_*^2 \frac{\partial u}{\partial \acute{z}} \\ &= \frac{U_*^3}{\kappa\,(\acute{z} + z_0)} \end{aligned} \tag{71}$$

In the E-ε model, ν_t is given by

$$\nu_t = C_\mu \frac{E^2}{\varepsilon}$$

Combining this with Equations (68) and (71) yields an expression for E:

$$E = \frac{U_*^2}{\sqrt{C_\mu}}$$

3.4 Summary and analysis

We therefore have an analytical solution to the zero-order equations, which can be summarized as:

$$U_0 = \frac{U_{*0}}{\kappa} \ln\left(\frac{\acute{z} + z_0}{z_0}\right) \tag{72}$$

$$p_0 = 0 \tag{73}$$

$$E_0 = \frac{U_{*0}^2}{\sqrt{C_\mu}} \tag{74}$$

$$\varepsilon_0 = \frac{U_{*0}^3}{\kappa(\acute{z} + z_0)} \tag{75}$$

$$\nu_0 = \kappa U_{*0}(\acute{z} + z_0) \tag{76}$$

Substituting these expressions back into the zero-order equations confirms that they constitute a valid solution, with one proviso, stemming from the substitution into the equation for ε_0: the constants of the $E - \varepsilon$ model must satisfy the relation

$$(C_{\varepsilon 2} - C_{\varepsilon 1})\sigma_\varepsilon \sqrt{C_\mu} = \kappa^2 \tag{77}$$

As a result of empirically fine-tuning the constants based on experience, the constants listed in Table 1 do not satisfy Equation (77) exactly. This is a further indication that the turbulence model is not based entirely on rigourous physics, but rather involves a significant empirical element.

In the present work, we assume that the zero-order solution derived in this section holds throughout the height of the domain. These expressions are thus substituted directly into the first-order equations wherever zero-order flow variables occur.

4 The first-order equations

The first-order equations are obtained by performing a perturbation expansion (Section 2.5) on the equations expressed in transformed coordinates (Section 2.4), and retaining only the first-order terms. We then proceed to structure them into ordinary differential equations of a form convenient for numerical integration:

1. All terms containing the unknown first-order flow variables are placed on the left-hand side (LHS) of the equations, and the other terms containing only zero-order flow variables (and known first-order geometry factors J_1, $\beta^i_{j(1)}$ and $\alpha^j_{i(1)}$) are grouped into the source terms S_i, S_E and S_ε on the right-hand side (RHS).
2. We Fourier-transform the equations horizontally.
3. The pressure and horizontal velocity components are eliminated from the momentum transport equation.
4. We derive a first-order equation for vorticity.
5. We formulate boundary conditions specifically for the first-order quantities.

The result is a set of four ordinary differential equations (ODEs), to be solved numerically. In the computer code, all the zero-order flow variables present in the first-order equations derived in this section are replaced by the expressions of the zero-order solution (Equations 72 to 76).

The full derivation and processing of the first-order equations is a somewhat lengthy affair involving a considerable amount of algebra, and therefore constitutes an open field for human error when done manually. In order to obtain an independent check on our manually-derived equations, we also performed the manipulations listed above using Mathematica (Wolfram Research, Inc. 2005), a specialized software package for symbolic manipulation. This allowed us to catch a few mistakes in our manual derivations (most notably, sign errors), but also in our Mathematica-based derivations. In the end, both methods arrived at the same result; we are thus confident of the mathematical correctness of the ODEs derived.

The full manual derivation of the first-order equations is presented in Appendix B. Only a summary of the results are presented here. The continuity equation is examined first, as it is most simple. We then move on to the TKE and dissipation equations; as we will see, their expressions will be required in the momentum equation, due to the term involving the first-order viscosity, ν_1.

We start, however, by introducing a shorthand notation that will help reduce the clutter in the final equations.

Shorthand notation

$$
\begin{array}{lcllcllcl}
\acute{z} & = & \acute{x}_3 & \mathcal{E} & = & \hat{E}_1 & \Omega & = & \hat{\omega}_3^{(1)} = \mathrm{i}\check{k}_q \hat{U}_q^{(1)} \\
\mathcal{W} & = & \hat{U}_3^{(1)} & \mathcal{E}' & = & \dfrac{\partial \mathcal{E}}{\partial \acute{z}} & \Omega' & = & \dfrac{\partial \Omega}{\partial \acute{z}} \\
\mathcal{W}' & = & \dfrac{\partial \mathcal{W}}{\partial \acute{z}} & \mathcal{E}'' & = & \dfrac{\partial^2 \mathcal{E}}{\partial \acute{z}^2} & \Omega'' & = & \dfrac{\partial \Omega}{\partial \acute{z}} \\
\mathcal{W}'' & = & \dfrac{\partial^2 \mathcal{W}}{\partial \acute{z}^2} & \mathcal{D} & = & \hat{\varepsilon}_1 & & & \\
\mathcal{W}''' & = & \dfrac{\partial^3 \mathcal{W}}{\partial \acute{z}^3} & \mathcal{D}' & = & \dfrac{\partial \mathcal{D}}{\partial \acute{z}} & A & = & \mathrm{i}k_q \acute{U}_q^{(0)} \\
\mathcal{W}'''' & = & \dfrac{\partial^4 \mathcal{W}}{\partial \acute{z}^4} & \mathcal{D}'' & = & \dfrac{\partial^2 \mathcal{D}}{\partial \acute{z}^2} & B & = & \mathrm{i}\check{k}_q \acute{U}_q^{(0)}
\end{array}
$$

for $q = 1, 2$. In B, we use the perpendicular wavenumber vector $(\breve{k}_1, \breve{k}_2) = (-k_2, k_1)$.

4.1 Eddy viscosity

The first-order viscosity is given by

$$\nu_1 = \nu_0 \left(2\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0} \right) \tag{78}$$

Using the shorthand notation, its Fourier-transformed form is

$$\hat{\nu}_1 = \nu_0 \left(2\frac{\mathcal{E}}{E_0} - \frac{\mathcal{D}}{\varepsilon_0} \right) \tag{79}$$

In subsequent equations, the first-order viscosity will be eliminated using this expression.

4.2 Continuity equation

We have simply

$$\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_j} = 0 \tag{80}$$

Fourier-transforming the continuity equation yields

$$\mathcal{F}\left\{ \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_j} \right\} = \mathcal{F}\left\{ \frac{\partial \acute{U}_1^{(1)}}{\partial \acute{x}_1} + \frac{\partial \acute{U}_2^{(1)}}{\partial \acute{x}_2} + \frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_3} \right\} = \mathrm{i}k_1 \hat{\acute{U}}_1^{(1)} + \mathrm{i}k_2 \hat{\acute{U}}_2^{(1)} + \frac{\partial \hat{\acute{U}}_3^{(1)}}{\partial \acute{x}_3} = 0$$

which can be rearranged into the simple form

$$\mathrm{i}k_q \hat{\acute{U}}_q^{(1)} = -\mathcal{W}' \tag{81}$$

with $q = 1, 2$. This expression will subsequently be substituted into the momentum transport equation in order to eliminate the horizontal velocity components.

4.3 TKE equation

The first-order TKE equation is:

$$\begin{aligned}
& \frac{\partial(\acute{U}_i^{(1)} E_0)}{\partial \acute{x}_i} + \frac{\partial(\acute{U}_i^{(0)} E_1)}{\partial \acute{x}_i} \\
& -\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_1}{\sigma_E} \frac{\partial E_0}{\partial \acute{x}_3} \right) \\
& -\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E} \frac{\partial E_1}{\partial \acute{x}_j} \right) \\
& -\left(2\,\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0} \right) \Pi_0 \\
& -2\,\nu_0 \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3} + \frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i} \right) \\
& +\varepsilon_1 \\
= \; & S_E
\end{aligned} \tag{82}$$

where

$$
\begin{aligned}
S_E \;=\;& \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_E}\alpha^p_{3(1)}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
&+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_j}\Big(J_1 E_0\Big)\right) \\
&+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha^q_{j(1)}E_0\Big)\right) \\
&-J_1\Pi_0 \\
&+2\,\nu_0\left(\frac{\partial \acute{U}^{(0)}_i}{\partial \acute{x}_j}+\frac{\partial \acute{U}^{(0)}_j}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}^{(0)}_i\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{r(1)}\acute{U}^{(0)}_r\big)\right] \\
&-J_1\varepsilon_0
\end{aligned}
\tag{83}
$$

After Fourier transformation and use of the shorthand notation, we get:

$$
\begin{aligned}
\frac{\partial \mathcal{E}'}{\partial \acute{z}} = \frac{\sigma_E}{\nu_0}\Bigg[& \\
&+\left[-\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right)-\frac{1}{\sigma_E}\frac{\partial \nu_0}{\partial \acute{z}}\right]\mathcal{E}' \\
&+\left[A-\frac{\partial}{\partial \acute{z}}\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right)+k_q k_q\frac{\nu_0}{\sigma_E}-2\,\frac{\Pi_0}{E_0}\right]\mathcal{E} \\
&+\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right)\mathcal{D}' \\
&+\left[\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right)+\frac{\Pi_0}{\varepsilon_0}+1\right]\mathcal{D} \\
&-2\,\nu_0\frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{z}}\mathcal{W}'' \\
&+\left[\frac{\partial E_0}{\partial \acute{z}}-2\,\nu_0\frac{\partial A}{\partial \acute{z}}\right]\mathcal{W} \\
&+2\,\nu_0\frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{z}}\Omega' \\
&-\hat{S}_E\Bigg]
\end{aligned}
\tag{84}
$$

with $q = 1, 2$, and where $\hat{S}_E$ is the Fourier-transformed source term.

4.4 Dissipation equation

The first-order dissipation equation is:

$$
\begin{aligned}
&\frac{\partial(\acute{U}_j^{(1)}\varepsilon_0)}{\partial\acute{x}_j}+\frac{\partial(\acute{U}_j^{(0)}\varepsilon_1)}{\partial\acute{x}_j}\\
&-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)\\
&-\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\varepsilon_1}{\partial\acute{x}_j}\right)\\
&-C_{\varepsilon1}C_\mu\frac{E_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2\\
&-2\,C_{\varepsilon1}C_\mu E_0\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_i}\right)\\
&+C_{\varepsilon2}\left(\frac{2\varepsilon_0\varepsilon_1}{E_0}-(\varepsilon_0)^2\frac{E_1}{(E_0)^2}\right)\\
=\;& S_\varepsilon
\end{aligned}
\tag{85}
$$

where

$$
\begin{aligned}
S_\varepsilon \;=\;& \frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu_0}{\sigma_\varepsilon}\alpha^p_{3(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)\\
&+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_j}\Big(J_1\varepsilon_0\Big)\right)\\
&+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_q}\Big(\alpha^q_{j(1)}\varepsilon_0\Big)\right)\\
&-C_{\varepsilon1}C_\mu\frac{E_0J_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2\\
&+2\,C_{\varepsilon1}C_\mu E_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)\left[\frac{\partial}{\partial\acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial\acute{x}_j}\big(\beta^i_{r(1)}\acute{U}_r^{(0)}\big)\right]\\
&-C_{\varepsilon2}J_1\frac{(\varepsilon_0)^2}{E_0}
\end{aligned}
\tag{86}
$$

After Fourier transformation and use of the shorthand notation, we get:

$$
\begin{aligned}
\frac{\partial \mathcal{D}'}{\partial \acute{z}} &= \frac{\sigma_\varepsilon}{\nu_0}\Bigg[\\
&+\frac{1}{\sigma_\varepsilon}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}} - \frac{\partial \nu_0}{\partial \acute{z}}\right)\mathcal{D}' \\
&+\left[A + \frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) + C_{\varepsilon 2}\frac{2\varepsilon_0}{E_0} + k_q k_q \frac{\nu_0}{\sigma_\varepsilon}\right]\mathcal{D} \\
&-2\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right)\mathcal{E}' \\
&+\left[-2\frac{\partial}{\partial \acute{z}}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) - \frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon 1}\Pi_0 - C_{\varepsilon 2}\left(\frac{\varepsilon_0}{E_0}\right)^2\right]\mathcal{E} \\
&-2\,C_{\varepsilon 1}C_\mu E_0 \frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{z}}\mathcal{W}'' + \left[\frac{\partial \varepsilon_0}{\partial \acute{z}} - 2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial A}{\partial \acute{z}}\right]\mathcal{W} \\
&+2\,C_{\varepsilon 1}C_\mu E_0 \frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{z}}\Omega' \\
&-\hat{S}_\varepsilon\Bigg]
\end{aligned}
\tag{87}
$$

with $q = 1, 2$, and where $\hat{S}_\varepsilon$ is the Fourier-transformed source term.

4.5 Momentum transport equation

The first-order momentum transport equation is

$$
\begin{aligned}
&\acute{U}_j^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \acute{U}_j^{(0)}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} - \frac{\partial}{\partial \acute{x}_j}\left(\nu_0\Big(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\Big)\right) \\
&\qquad -\frac{\partial}{\partial \acute{x}_j}\left(\nu_1\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) + \frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_i} \;=\; S_i
\end{aligned}
\tag{88}
$$

where

$$
\begin{aligned}
S_i \;=\;& \acute{U}_j^{(0)}\frac{\partial}{\partial \acute{x}_j}\Big(J_1 \acute{U}_i^{(0)} - \beta^i_{q(1)}\acute{U}_q^{(0)}\Big) + \frac{\partial}{\partial \acute{x}_q}\left(\alpha^q_{j(1)}\nu_0\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) \\
&+\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\Big(\frac{\partial}{\partial \acute{x}_r}\big(\alpha^r_{j(1)}\acute{U}_i^{(0)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{p(1)}\acute{U}_p^{(0)}\big) + \frac{\partial}{\partial \acute{x}_r}\big(\alpha^r_{i(1)}\acute{U}_j^{(0)}\big) + \frac{\partial}{\partial \acute{x}_i}\big(\beta^j_{p(1)}\acute{U}_p^{(0)}\big)\Big)\right)
\end{aligned}
\tag{89}
$$

Fourier-transformation of the momentum transport equation

In the case $i = 1, 2$:

$$
\begin{aligned}
&\hat{\acute{U}}_3^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3} + \mathsf{i}\left(k_1\acute{U}_1^{(0)} + k_2\acute{U}_2^{(0)}\right)\hat{\acute{U}}_i^{(1)} \\
&+\left(k_1 k_1 + k_2 k_2\right)\nu_0\hat{\acute{U}}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial \hat{\acute{U}}_i^{(1)}}{\partial \acute{x}_3}\right) \\
&-\mathsf{i}k_i\frac{\partial \nu_0}{\partial \acute{x}_3}\hat{\acute{U}}_3^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right) + \frac{1}{\rho}\mathsf{i}k_i\hat{p}_1 \;=\; \hat{S}_i
\end{aligned}
\tag{90}
$$

and in the case $i = 3$:

$$\mathsf{i}\left(k_1\acute{U}_1^{(0)} + k_2\acute{U}_2^{(0)}\right)\hat{\acute{U}}_3^{(1)} + (k_1k_1 + k_2k_2)\,\nu_0\hat{\acute{U}}_3^{(1)} - \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial\hat{\acute{U}}_3^{(1)}}{\partial\acute{x}_3}\right)$$
$$-\frac{\partial\nu_0}{\partial\acute{x}_3}\frac{\partial\hat{\acute{U}}_3^{(1)}}{\partial\acute{x}_i} - \mathsf{i}\hat{\nu}_1\left(k_1\frac{\partial\acute{U}_1^{(0)}}{\partial\acute{x}_3} + k_2\frac{\partial\acute{U}_2^{(0)}}{\partial\acute{x}_3}\right) + \frac{1}{\rho}\frac{\partial\hat{p}_1}{\partial\acute{x}_3} = \hat{S}_3 \qquad (91)$$

Source term

$$\begin{aligned}
\hat{S}_q \;=\;& \mathsf{i}k_r\acute{U}_r^{(0)}\left(\hat{J}_1\acute{U}_q^{(0)} - \hat{\beta}^q_{r(1)}\acute{U}_r^{(0)}\right)\\
&+\mathsf{i}k_r\hat{\alpha}^r_{3(1)}\nu_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\right)\\
&-k_sk_r\nu_0\hat{\alpha}^r_{s(1)}\acute{U}_q^{(0)} + \mathsf{i}k_s\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{s(1)}\acute{U}_q^{(0)}\right)\\
&+\mathsf{i}k_r\frac{\partial}{\partial\acute{x}_3}\left(\nu_0\hat{\alpha}^r_{3(1)}\acute{U}_q^{(0)}\right) + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_q^{(0)}\right)\right)\\
&-k_rk_r\nu_0\hat{\beta}^q_{s(1)}\acute{U}_s^{(0)} + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^q_{s(1)}\acute{U}_s^{(0)}\right)\right)\\
&-k_rk_s\nu_0\hat{\alpha}^s_{q(1)}\acute{U}_r^{(0)} + \mathsf{i}k_r\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{q(1)}\acute{U}_r^{(0)}\right)\\
&-k_rk_q\nu_0\hat{\beta}^r_{s(1)}\acute{U}_s^{(0)} + \mathsf{i}k_q\frac{\partial}{\partial\acute{x}_3}\left(\nu_0\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)
\end{aligned} \qquad (92)$$

where $q, r, s = 1, 2$ and

$$\begin{aligned}
\hat{S}_3 \;=\;& -\mathsf{i}k_r\acute{U}_r^{(0)}\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\\
&+\mathsf{i}k_r\hat{\alpha}^r_{s(1)}\nu_0\frac{\partial\acute{U}_s^{(0)}}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{s(1)}\nu_0\frac{\partial\acute{U}_s^{(0)}}{\partial\acute{x}_3}\right)\\
&-k_rk_r\nu_0\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)} + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)\right)\\
&-k_sk_r\nu_0\hat{\alpha}^r_{3(1)}\acute{U}_s^{(0)} + \mathsf{i}k_s\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_s^{(0)}\right)\\
&+\mathsf{i}k_r\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^r_{s(1)}\acute{U}_s^{(0)}\right) + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)\right)
\end{aligned} \qquad (93)$$

where $r, s = 1, 2$

Eliminating the pressure and horizontal velocity components

We wish to eliminate p_1 as well as $\acute{U}_1^{(1)}$ and $\acute{U}_2^{(1)}$ from the first-order equations, so that only an equation for $\acute{U}_3^{(1)}$ remains.

This is done by substituting the continuity equation (81) into the horizontal momentum equation. We then take the vertical derivative of the horizontal momentum equation and add it to the vertical momentum equation. After a great deal of rearranging, we obtain

$$
\begin{aligned}
\frac{\partial \mathcal{W}'''}{\partial \acute{z}} = -\frac{1}{\nu_0}\Bigg[&+ 2\,\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}''' \\
&+ \left(-A - 2\,k_j k_j \nu_0 + \frac{\partial^2 \nu_0}{\partial \acute{z}^2}\right)\mathcal{W}'' \\
&- 2\,k_j k_j \frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}' \\
&+ \left(\frac{\partial^2 A}{\partial \acute{z}^2} + k_j k_j A + k_j k_j k_q k_q \nu_0 + k_j k_j \frac{\partial^2 \nu_0}{\partial \acute{z}^2}\right)\mathcal{W} \\
&- 2\,\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\mathcal{E}'' \\
&- 4\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right)\mathcal{E}' \\
&+ \left[-2\,\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right) - 2\,k_j k_j \frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right]\mathcal{E} \\
&+ \frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\mathcal{D}'' \\
&+ 2\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right)\mathcal{D}' \\
&+ \left[\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right) + k_j k_j \frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right]\mathcal{D} \\
&- \mathsf{i}k_j \frac{\partial \hat{S}_j}{\partial \acute{z}} \\
&- k_j k_j \hat{S}_3\Bigg]
\end{aligned}
\tag{94}
$$

But $\mathcal{E}'' = \frac{\partial \mathcal{E}'}{\partial z}$ and $\mathcal{D}'' = \frac{\partial \mathcal{D}'}{\partial z}$, and Equations (84) and (87) precisely provide expressions for these. We therefore substitute them for $\mathcal{E}''$ and $\mathcal{D}''$ in the code.

4.6 Obtaining an equation for first-order vorticity

As seen above, the expression $\mathsf{i}\check{k}_q\hat{U}_q^{(1)}$ appears in the production terms of the first-order TKE and dissipation equations. This expression is simply the first-order vertical vorticity, $\acute{\omega}_3^{(1)}$, expressed in the Fourier domain:

$$
\begin{aligned}
\mathsf{i}\check{k}_q\hat{U}_q^{(1)} &= \mathsf{i}\check{k}_1\hat{U}_1^{(1)} + \mathsf{i}\check{k}_1\hat{U}_1^{(1)} \\
&= -\mathsf{i}k_2\hat{U}_1^{(1)} + \mathsf{i}k_2\hat{U}_1^{(1)} \\
&= \mathcal{F}\left\{-\frac{\partial \hat{U}_1^{(1)}}{\partial \acute{x}_2} + \frac{\partial \hat{U}_1^{(1)}}{\partial \acute{x}_1}\right\} \\
&= \mathcal{F}\left\{\acute{\omega}_3^{(1)}\right\} = \hat{\acute{\omega}}_3^{(1)} = \Omega
\end{aligned}
\tag{95}
$$

The vorticity Ω cannot be expressed in terms of the variables $\mathcal{W}, \mathcal{E}, \mathcal{D}$ and their derivatives. We must therefore derive an equation for Ω. We do this by multiplying the $\hat{U}_q^{(1)}$ equation by $\mathsf{i}\check{k}_q$ $(q = 1, 2)$. Doing so yields

$$\frac{\partial \Omega'}{\partial \acute{z}} = \frac{1}{\nu_0}\Bigg[-\frac{\partial \nu_0}{\partial \acute{z}}\Omega' + (B + k_r k_r \nu_0)\,\Omega + \frac{\partial B}{\partial \acute{z}}\mathcal{W} - 2\,\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{z}}\mathcal{E}' - 2\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{z}}\right)\mathcal{E} - \frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{z}}\mathcal{D}' - \frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{z}}\right)\mathcal{D} - \mathrm{i}\breve{k}_q \hat{S}_q\Bigg] \tag{96}$$

4.7 First-order pressure

Recall that $p_0 = 0$ (Equation 73), so that $p = p_1$. Once the equations are solved, the pressure is obtained from

$$\begin{aligned}
\rho\hat{p} \;=\; & -4\mathrm{i}\hat{h}_\mathrm{t} U_{*0}^2 k e^{-k\acute{z}} - 2\mathrm{i}\hat{h}_\mathrm{t} z_0 U_{*0}^2 k^2 e^{-k\acute{z}} + 2\mathrm{i}\hat{h}_\mathrm{t} U_{*0}^2 k^2 (z_0+\acute{z}) e^{-k\acute{z}} + \frac{\mathrm{i}\kappa\mathcal{D}}{U_{*0}k} \\
& + \frac{\mathrm{i}\kappa(z_0+\acute{z})\mathcal{D}'}{U_{*0}k} - 2\mathrm{i}\sqrt{C_\mu}\mathcal{E}' k^{-1} - 2\mathrm{i}\hat{h}_\mathrm{t} z_0 U_{*0}^2 k^2 \ln\left(\frac{z_0+\acute{z}}{z_0}\right) e^{-k\acute{z}} \\
& + 4\mathrm{i}\hat{h}_\mathrm{t} U_{*0}^2 k^2 (z_0+\acute{z}) \ln\left(\frac{z_0+\acute{z}}{z_0}\right) e^{-k\acute{z}} + \hat{h}_\mathrm{t} z_0 U_{*0}^2 k^2 \left[\ln\left(\frac{z_0+\acute{z}}{z_0}\right)\right]^2 e^{-k\acute{z}}\kappa^{-2} \\
& - \hat{h}_\mathrm{t} U_{*0}^2 k^2 (z_0+\acute{z}) \left[\ln\left(\frac{z_0+\acute{z}}{z_0}\right)\right]^2 e^{-k\acute{z}}\kappa^{-2} + \kappa U_{*0}\mathcal{W} + \frac{\mathrm{i}U_{*0}\mathcal{W}}{\kappa k(z_0+\acute{z})} \\
& - \kappa U_{*0}(z_0+\acute{z})\mathcal{W}' - \mathrm{i}U_{*0}\ln\left(\frac{z_0+\acute{z}}{z_0}\right)\mathcal{W}'\kappa^{-1}k^{-1} \\
& + \kappa U_{*0}\mathcal{W}''k^{-2} + \kappa U_{*0}(z_0+\acute{z})\mathcal{W}'''k^{-2}
\end{aligned} \tag{97}$$

4.8 Boundary conditions for first-order quantities

The boundary conditions (BCs) presented in Section 2.7 must be expanded so as to provide boundary conditions specifically for the first-order variables.

Lower BCs

Equations (45), (46), (47), and (49) simply translate as, respectively,

$$\begin{aligned}
\mathcal{W} &= 0 \\
\mathcal{W}' &= 0 \\
\mathcal{E}' &= 0 \\
\Omega &= 0
\end{aligned}$$

The RHS of Equation (48) must be expanded. This results in

$$\mathcal{D} - \frac{3}{2}\frac{C_\mu^{\frac{3}{4}}}{\kappa z_0} E_0^{\frac{1}{2}} \mathcal{E} = 0$$

We therefore have $n_1 = 4$ lower BCs ($n_1 = 5$ in the 3D case). The freely specifiable variables are $\mathcal{W}'', \mathcal{W}''', \mathcal{E}, \mathcal{D}'$ as well as Ω' in the 3D case.

Upper BCs

We assume that the coordinate transformation vanishes at the upper boundary, hence the coordinate system is back to Cartesian. The first-order geometry factors thus vanish ($J_1 = \beta^i_{j(1)} = \alpha^j_{i(1)} = 0$) and we have $u_i = \acute{U}_i$. The BCs in Equations (50), (52), and (54) simply translate as, respectively,

$$\begin{aligned} \mathcal{W} &= 0 \\ \mathcal{E}' &= 0 \\ \Omega &= 0 \end{aligned}$$

Since $u_i = \acute{U}_i$, the top BC for momentum flux (Equation 56) can be written

$$C_\mu \frac{E^2}{\varepsilon} \frac{\partial \acute{U}_i}{\partial \acute{z}} = U_*^2 e_i$$

where e_i is the unit vector along the wind direction. Substituting the analytical expressions of the zero-order solution into the left-hand side, we get

$$C_\mu \frac{U_{*0}^4/C_\mu}{U_{*0}^3/(\kappa(z+z_0))} \frac{U_{*0} e_i}{\kappa(z+z_0)} = U_{*0}^2 e_i$$

and we see that the zero-order solution satisfies the BC on its own. Therefore the contribution of the first-order equations to this BC must be equal to zero:

$$\left[C_\mu \frac{E^2}{\varepsilon} \frac{\partial \acute{U}_q}{\partial \acute{z}} \right]_1 = 0$$

i.e.

$$C_\mu \frac{E_0^2}{\varepsilon_0} \frac{\partial \acute{U}_q^{(1)}}{\partial \acute{z}} + C_\mu \left(2\frac{E_1 E_0}{\varepsilon_0} - \frac{E_0^2 \varepsilon_1}{\varepsilon_0^2} \right) \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{z}} = 0$$

Fourier-transforming, multiplying by $\mathrm{i}k_q$, and using continuity reduces the equation to

$$\mathcal{W}'' = 0 \tag{98}$$

The first-order version of Equation (51) is $\hat{\varepsilon}_1 = \hat{\Pi}_1$. In the full derivation of the first-order equation (Equation B.127), we saw that the first-order production term is

$$\begin{aligned} \Pi_1 &= \left(2\,\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0} - J_1 \right) \Pi_0 \\ &\quad + 2\,\nu_0 \left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i} \right) \left[\frac{\partial}{\partial \acute{x}_p} \left(\alpha^p_{j(1)} \acute{U}_i^{(0)} \right) + \frac{\partial}{\partial \acute{x}_j} \left(\beta^i_{r(1)} \acute{U}_r^{(0)} \right) + \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} \right] \\ &= \left(2\,\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0} \right) \Pi_0 + 2\,\nu_0 \left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i} \right) \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} \end{aligned}$$

where we used $\alpha^p_{j(1)} = \beta^i_{r(1)} = J_1 = 0$ at the upper boundary. Fourier-transforming, using the previously derived result

$$\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i} \right) \frac{\partial \hat{\acute{U}}_i^{(1)}}{\partial \acute{x}_j} = \frac{1}{k_r k_r} \left(\frac{\partial A}{\partial \acute{z}} \mathcal{W}'' - \frac{\partial B}{\partial \acute{z}} \Omega' \right) + \frac{\partial A}{\partial \acute{z}} \mathcal{W}$$

and noting that $\Pi_0 = \varepsilon_0$ and that $\mathcal{W} = 0$ and $\mathcal{W}'' = 0$ at the upper boundary, we can write the boundary condition $\mathcal{D} = \hat{\Pi}_1$ as

$$\mathcal{D} - \frac{\varepsilon_0}{E_0} \mathcal{E} + \frac{\nu_0}{k_r k_r} \frac{\partial B}{\partial \acute{z}} \Omega' = 0$$

We therefore have $n_2 = 4$ lower BCs ($n_2 = 5$ in the 3D case). The freely specifiable variables are $\mathcal{W}', \mathcal{W}''', \mathcal{E}, \mathcal{D}'$ as well as Ω' in the 3D case.

Matrix form of the 2D boundary conditions

In the 2D case, there is no vorticity ($\Omega = \Omega' = 0$). The lower and upper BCs can be written in matrix form as

$$\mathbf{MX} = 0 \qquad \text{at} \qquad \acute{z} = 0 \tag{99}$$

$$\mathbf{QX} = 0 \qquad \text{at} \qquad \acute{z} = \acute{z}_{\text{top}} \tag{100}$$

where

$$\mathbf{M} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{3}{2}\frac{C_\mu^{\frac{3}{4}}}{\kappa \acute{z}_0} E_0^{\frac{1}{2}} & 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{Q} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & -\frac{\varepsilon_0}{E_0} & 0 & 1 & 0 \end{pmatrix}$$

$$\mathbf{X} = \begin{pmatrix} \mathcal{W} \\ \mathcal{W}' \\ \mathcal{W}'' \\ \mathcal{W}''' \\ \mathcal{E} \\ \mathcal{E}' \\ \mathcal{D} \\ \mathcal{D}' \end{pmatrix}$$

5 Solution technique

In this chapter, we present the techniques used to solve the first-order equations, and discuss the associated numerical difficulties.

5.1 Reduction to a set of first-order ODEs

Equations (84), (87), (94) and (96) constitute the system to be solved:

- one fourth-order[1] ordinary differential equation (ODE) for $\mathcal{W}$, and
- one second-order ODE for each of $\mathcal{E}$, $\mathcal{D}$ and Ω.

This system can be reduced to a set of ten first-order ODEs, by treating $\mathcal{W}$, $\mathcal{W}'$, $\mathcal{W}''$, $\mathcal{W}'''$, $\mathcal{E}$, $\mathcal{E}'$, $\mathcal{D}$, $\mathcal{D}'$, Ω, and Ω' as ten different, dependent variables, and $\acute{z}$ as the only independent variable, and by using the following definitions as six additional equations:

$$\left.\begin{aligned} \frac{\partial \mathcal{W}}{\partial \acute{z}} &= \mathcal{W}' \\ \frac{\partial \mathcal{W}'}{\partial \acute{z}^2} &= \mathcal{W}'' \\ \frac{\partial \mathcal{W}''}{\partial \acute{z}^3} &= \mathcal{W}''' \\ \frac{\partial \mathcal{E}}{\partial \acute{z}} &= \mathcal{E}' \\ \frac{\partial \mathcal{D}}{\partial \acute{z}} &= \mathcal{D}' \\ \frac{\partial \Omega}{\partial \acute{z}} &= \Omega' \end{aligned}\right\} \tag{101}$$

This system of linear, first-order ODEs can be written in matrix form as

$$\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{A}\mathbf{X} + \mathbf{S} \tag{102}$$

where, in the 2D case,

$$\mathbf{X} = \begin{pmatrix} \mathcal{W} \\ \mathcal{W}' \\ \mathcal{W}'' \\ \mathcal{W}''' \\ \mathcal{E} \\ \mathcal{E}' \\ \mathcal{D} \\ \mathcal{D}' \end{pmatrix} \quad \text{and the source term} \quad \mathbf{S} = \begin{pmatrix} 0 \\ 0 \\ 0 \\ -\mathrm{i}k\frac{\partial \hat{S}_1}{\partial \acute{z}} - k^2 \hat{S}_3 \\ 0 \\ -\hat{S}_E \\ 0 \\ -\hat{S}_\varepsilon \end{pmatrix}$$

and $\mathbf{A}$ is the 8×8 matrix containing the coefficients of Equations (84), (87), (94), and the first five equations in (101). The matrices $\mathbf{X}$, $\mathbf{A}$ and $\mathbf{S}$ are all functions of $\acute{z}$. The source term $\mathbf{S}$ contains the factor $e^{-k\acute{z}}$ and therefore dies quickly as $\acute{z} \to \infty$.

In the 3D case, Ω and Ω' and their equations would be included. However, except for a few remarks, we will focus on the 2D case for the remainder of this document, as only the 2D case has been implemented.

[1] Note that "order" refers here to the formal order of the ODE, i.e. the order of its highest derivative in $\acute{z}$. This is not to be confused with the "zero-order" and "first-order" terminology used elsewhere in this document, referring to the horizontally homogeneous flow and the orography-induced perturbation, respectively.

5.2 Scaling and non-dimensionalization

The system of linear ODEs in Equation (102) can be conveniently scaled such that its solution in one particular case can also be applied to other cases. This will now be examined.

Scaling by terrain height

Equation (102) still holds if both $\mathbf{X}$ and $\mathbf{S}$ are scaled by some arbitrary function $\hat{f}(\mathbf{k})$ independent of $\acute{z}$. Upon examining the elements of the source term matrix $\mathbf{S}$, we note that they all include the factor $\hat{h}_{\mathrm{t}}(\mathbf{k})$, ultimately because the first-order metrics $\hat{\alpha}^i_{j(1)}$ and $\hat{\beta}^j_{i(1)}$ that they contain are scaled by the terrain height $\hat{h}_{\mathrm{t}}(\mathbf{k})$ found in the coordinate transformation function $\Lambda_j(\mathbf{k}, \acute{z})$ (see Section 2.2). We make use of this in the scaling described below.

Non-dimensionalization

Let us define the length scale L_σ and the time scale T_σ

$$\begin{aligned} L_\sigma &= \frac{1}{k} \\ T_\sigma &= \frac{L_\sigma}{U_{*0}} \end{aligned}$$

and, using those scales as well as $\hat{h}_{\mathrm{t}}(k)$, the non-dimensional variables

$$\begin{aligned} \tilde{\acute{z}} &= \acute{z} L_\sigma^{-1} \\ \tilde{\mathcal{W}} &= \mathcal{W} T_\sigma \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{W}}' &= \mathcal{W}' L_\sigma T_\sigma \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{W}}'' &= \mathcal{W}'' L_\sigma^2 T_\sigma \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{W}}''' &= \mathcal{W}''' L_\sigma^3 T_\sigma \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{E}} &= \mathcal{E} L_\sigma^{-1} T_\sigma^2 \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{E}}' &= \mathcal{E}' T_\sigma^2 \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{D}} &= \mathcal{D} L_\sigma^{-1} T_\sigma^3 \hat{h}_{\mathrm{t}}(k)^{-1} \\ \tilde{\mathcal{D}}' &= \mathcal{D}' T_\sigma^3 \hat{h}_{\mathrm{t}}(k)^{-1} \end{aligned}$$

Substituting these into the equations in place of the dimensional variables yields (after some algebra) a system of ODEs,

$$\frac{\partial \tilde{\mathbf{X}}}{\partial \tilde{\acute{z}}} = \tilde{\mathbf{A}} \tilde{\mathbf{X}} + \tilde{\mathbf{S}}$$

depending on only two non-dimensional parameters:

- $k\acute{z} = \acute{z}/L_\sigma$, a non-dimensional height, and
- $kz_0 = z_0/L_\sigma$, a non-dimensional roughness.

As we will now see, this presents an formidable opportunity to address one of the main requirements posed to the user end our flow model, namely quickness.

Generation and usage of look-up tables

We exploit the dependency of our system of ODEs on only two non-dimensional parameters by solving the equations for $\tilde{\mathbf{X}}$ all foreseeable combinations $(k\acute{z}, kz_0)$

of these parameters, and storing the results into look-up tables (LUT). This is a one-time procedure, during which all the computationally heavy work is done.

From the user end, the flow field over any given orography with roughness z_0 is then quickly obtained by:

1. Fourier-transforming the orography;
2. For each wavenumber component k and each height of interest $\acute{z}$,
   (a) interpolating from the LUTs the values of the dependent variables in $\tilde{\mathbf{X}}$,
   (b) scaling the variables to obtain $\mathbf{X}$;
3. Returning to the real domain by inverse Fourier transform

In the scaling process, the friction velocity U_{*0} is set so that the calculated velocity matches a measured velocity at some point in the domain. This is an easy task, since $\mathcal{W}$, $\acute{U}_i$, and ultimately u_j, are linearly proportional to $T_\sigma^{-1} \propto U_{*0}$.

This is the procedure that is executed by the user-end portion of RAMSIM, representing a relatively light computational cost.

Range of $k\acute{z}$ and kz_0 As mentioned in Section 2.7, the calculation domain spans from $k\acute{z} = 0$ at the lower boundary to $k\acute{z}_{\mathrm{top}} = 10$ at the upper boundary.

As for the non-dimensional roughness kz_0, its maximum foreseeable range spans at least nine decades, as summarized in Table 2. We observed that the dependency of the variables $\mathcal{W}, \mathcal{E}, \mathcal{D}$ on kz_0 was rather smooth; 10^2 samples of kz_0 spread across this range were more than sufficient to generate a proper LUT.

Table 2: Range of the non-dimensional roughness parameter kz_0

Terrain feature	Example	k (m^{-1})	z_0 (m)	kz_0
Long and smooth	20-km snow surface	3.14×10^{-4}	2×10^{-4}	6.28×10^{-8}
Short and rough	1-m bump in a forest	6.28	1	6.28

Remarks on scaling in the 3D case

The 3D equations depend on an additional parameter: the angle γ between the wavenumber vector and the direction of the forcing stress at the upper boundary. Hence, in the scaling process, both U_{*0} and the azimuthal angle (relative to North) of the forcing stress θ_{f} must be set so that the calculations match a measured wind speed u and direction θ_{m} (relative to North) at some point in the domain. Whereas u_j is simply linearly proportional to U_{*0}, the relation between θ_{f} and the wind direction θ_{m} at an arbitrary point in the domain cannot be determined in advance: orographic features can deflect the wind in ways that depend on their own shape as well as on θ_{f}. It may therefore be necessary to find the appropriate θ_{f} by iteration.

If one is trying to match one specific measurement θ_{m}, then the total computation time is multiplied by the number of iterations. With a $\theta_{\mathrm{f}} = \theta_{\mathrm{m}}$ initial condition (still neglecting the Coriolis force), three or four iterations would presumably suffice.

In the context of wind resource assessment, however, the increase in computational expense is much less, since each iteration yields useful information. One wishes to calculate the flow field for wind coming from all directions θ_{m}, binned

into, say, twelve 30°-sectors (see Section 1.3). Given an unknown relationship between θ_f and θ_m, the first, best guess is to sample θ_f at 12 equal intervals, once in each 30°-sector. Note that obtaining the flow field for $n = 12$ different θ_f's uniformly distributed over 0°–360° requires only $n/4 = 3$ different calculations, due to the mirroring of the terrain, which we will describe on page 72.

One can then evaluate whether θ_m was sampled appropriately. This is illustrated for a hypothetical case in Figure 7. After sampling θ_f uniformly, the θ_m sectors 90°–

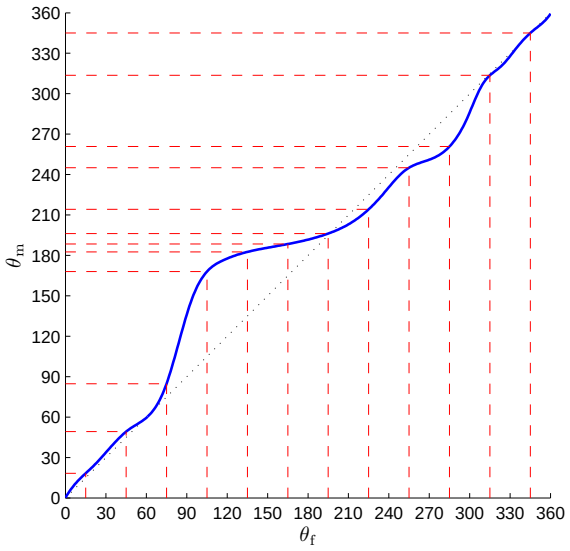


Figure 7: Sampling (red dashed lines) a hypothetical, unknown relationship (blue line) between the azimuthal angle of the upper-boundary forcing stress θ_f and the wind direction at a given point in the domain, θ_m.

150° and 270°–300° are under-sampled, and so one would add an extra calculation around $\theta_f = 300°$ and perhaps two more around $\theta_f = 90°$. Three calculations were required for the initial sampling, plus three more to correct deficiencies. So a roughly uniform sampling over twelve sectors of θ_m required a total of only six calculations. If a more uniform sampling of θ_m is needed, then a few more calculations can be performed. On the other hand, in certain lucky cases, perhaps the initial uniform sampling with three calculations will suffice, requiring no extra correction.

5.3 Shooting: Two-point boundary value problem

To avoid notational clutter, we will in this section (5.3) drop the tilde ˜ indicating non-dimensionality. Our task is to solve the system of N_v linear, first-order ODEs,

$$\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{A}\mathbf{X} + \mathbf{S} \tag{103}$$

subject to the boundary conditions derived in Section 4.8. Since we specify boundary conditions both at the lower (ground) and the upper (sky) boundaries, this constitutes a two-point boundary value problem. We solve it by numerical integration and the technique known as shooting to a fitting point. For non-linear ODEs, shooting is an iterative process; our linear ODEs, however, are solved by a more straightforward method, to which we now turn.

In the 2D case, we have $N_v = 8$ variables and equations; $n_1 = 4$ boundary conditions are specified at the lower boundary, and $n_2 = N_v - n_1 = 4$ at the upper boundary. In the 3D case, not examined here, the same methods as are described in the following can be applied, with $N_v = 10$, $n_1 = 5$, and $n_2 = 5$.

Linear combination of solutions

The source term $\mathbf{S}$ makes (103) an inhomogeneous equation. The corresponding homogeneous equation is

$$\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{A}\mathbf{X} \tag{104}$$

If $\mathbf{X}_{\text{tonic}}$ is a solution to the inhomogeneous equation (103) and $\mathbf{X}_{\text{gin}}$ is a solution to the homogeneous equation (104), then $\mathbf{X}_{\text{G\&T}} = \mathbf{X}_{\text{gin}} + \mathbf{X}_{\text{tonic}}$ is also a solution to the inhomogeneous equation — this is a tenet of ODE theory. In fact, one can "make it a double": $\mathbf{X}_{\text{tonic}} + \phi\mathbf{X}_{\text{gin}}$ is also a solution to the inhomogeneous equation for any value of the complex constant ϕ, since $\mathbf{X} = \phi\mathbf{X}_{\text{gin}}$ satisfies linear Equation (104) just as well as $\mathbf{X} = \mathbf{X}_{\text{gin}}$ does.

Why do we care? In order to integrate upwards from the lower boundary, one must supply a full set of N_{v} boundary values at the start of the integration. However, these cannot be determined uniquely from the n_1 available boundary conditions. One can supply an "arbitrary guess" at what the N_{v} starting values are. Upward integration using these starting boundary conditions will yield a trial solution $\mathbf{X}_0$ to the inhomogeneous equation (103) that does not, in general, satisfy the n_1 lower and n_2 upper boundary conditions.

If we then proceed to find n independent solutions $\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_n$ to the homogeneous equation (104), we can extend the trick described above to express a general solution $\mathbf{X}_{\text{inh}}$ to our inhomogeneous equation as

$$\mathbf{X}_{\text{inh}} = \mathbf{X}_0 + \mathbf{X}_{\text{hom}}\mathbf{\Phi} \tag{105}$$

where $\mathbf{X}_{\text{hom}} = (\mathbf{X}_1 \quad \mathbf{X}_2 \quad \ldots \quad \mathbf{X}_n)$ and $\mathbf{\Phi} = \begin{pmatrix} \phi_1 \\ \phi_2 \\ \ldots \\ \phi_n \end{pmatrix}$.

The secret is now to mix well: one must find the n weighting coefficients $\phi_1, \phi_2, \ldots, \phi_n$ that yield the linear combination satisfying all the boundary conditions. We do this using the shooting technique, of which there are two variants, which we now describe.

Shooting upwards

This variant is based on integrating from the lower boundary upwards using a variety of starting conditions. Alternatively, one may start at the upper boundary and shoot downwards; the method is exactly analogous.

We perform $n_2 + 1 = 5$ shots from the lower boundary, labelled $i = 0, \ldots, n_2$, each with a different starting condition $\mathbf{X}_i(0)$:

- Shot 0: The inhomogeneous equation is integrated using the starting condition in which all freely specifiable variables $(\mathcal{W}'', \mathcal{W}''', \mathcal{E}, \mathcal{D}')$ are set to zero, and the others variables are adjusted so that the lower BCs (Equation 99) are obeyed. This shot yields the solution $\mathbf{X}_0(\acute{z}_{\text{top}})$.
- Shots $i = 1, \ldots, n_2$: the *homogeneous* equation is integrated using the starting condition in which the i^{th} freely specifiable variables of the set $(\mathcal{W}'', \mathcal{W}''', \mathcal{E}, \mathcal{D}')$ is set to 1, and the three others to zero; and again, the others variables are adjusted so that the lower BCs are obeyed. These shots yield the solutions $\mathbf{X}_i$.

The "true" solution, i.e. satisfying all the BCs, can be expressed as a linear combination of the $n_2 + 1$ shots — as in Equation (105), with $n = n_2$. This true

solution necessarily satisfies the n_1 lower BCs, since each individual shot satisfies them.

Satisfaction of the upper BCs depends on the weighting factors ϕ_q. To find these, we substitute Equation (105) into Equation (100):

$$\mathbf{Q}\left(\mathbf{X}_0^{\text{top}} + \mathbf{X}_{\text{hom}}^{\text{top}}\mathbf{\Phi}\right) \quad = \quad 0$$

i.e.

$$(\mathbf{Q}\mathbf{X}_{\text{hom}}^{\text{top}})\mathbf{\Phi} \quad = \quad -\mathbf{Q}\mathbf{X}_0^{\text{top}} \tag{106}$$

This linear system is readily solved for $\mathbf{\Phi}$, thus yielding the weighting coefficients.

Shooting to a fitting point

Sometimes simple shooting is not possible: due to numerically difficult equations, the individual shots do not manage to reach the far boundary, or do so very unstably and become contaminated with numerical artifacts. In such a case, one can shoot to a fitting point located at a height $\acute{z}_{\text{fit}}$ judiciously placed somewhere within the domain. A total of $n_2 + 1$ shots are then fired from the lower boundary (n_2 shots with the homogeneous equations plus one with the inhomogeneous equations) and $n_1 + 1$ from the upper boundary.

The "true" solution in the lower part of the domain (below the fitting point) satisfying all the BCs, can be expressed as a linear combination of the $n_2 + 1$ trial solutions:

$$\mathbf{X}^{\text{bu}} = \mathbf{X}_0^{\text{bu}} + \mathbf{X}_{\text{hom}}^{\text{bu}}\mathbf{\Phi}^{\text{bu}} \tag{107}$$

where "bu" stands for "bottom-up". Similarly for the solution in the upper part of the domain,

$$\mathbf{X}^{\text{td}} = \mathbf{X}_0^{\text{td}} + \mathbf{X}_{\text{hom}}^{\text{td}}\mathbf{\Phi}^{\text{td}} \tag{108}$$

where "td" stands for "top-down". The requirement is that the solution be continuous, so that $\mathbf{X}^{\text{bu}} = \mathbf{X}^{\text{td}}$ at $\acute{z} = \acute{z}_{\text{fit}}$. From Equations (107) and (108), we therefore have

$$\mathbf{X}_0^{\text{bu}} + \mathbf{X}_{\text{hom}}^{\text{bu}}\mathbf{\Phi}^{\text{bu}} = \mathbf{X}_0^{\text{td}} + \mathbf{X}_{\text{hom}}^{\text{td}}\mathbf{\Phi}^{\text{td}} \quad \text{at} \quad \acute{z} = \acute{z}_{\text{fit}}$$

which is equivalent to

$$\begin{pmatrix} -\mathbf{X}_{\text{hom}}^{\text{bu}} & \mathbf{X}_{\text{hom}}^{\text{td}} \end{pmatrix} \begin{pmatrix} \mathbf{\Phi}^{\text{bu}} \\ \mathbf{\Phi}^{\text{td}} \end{pmatrix} = \mathbf{X}_0^{\text{bu}} - \mathbf{X}_0^{\text{td}} \quad \text{at} \quad \acute{z} = \acute{z}_{\text{fit}} \tag{109}$$

This linear system is then solved to find the unknown weighting factors $\mathbf{\Phi}^{\text{bu}}$ and $\mathbf{\Phi}^{\text{td}}$, which are used to obtain the solution in the lower and upper part of the domain from Equations (107) and Equation (108), respectively.

We used this method, and found that a fitting point placed at $k\acute{z} = 0.95$ minimized the numerical noise problem when $kz_0 \to 0$ (to be discussed on page 69).

5.4 Numerical integration

Remains to be discussed the numerical integration itself, i.e. the manner in which we obtain the individual "shots". The ODE we wish to integrate has the form

$$\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{f}(\acute{z}, \mathbf{X}) \tag{110}$$

where $\mathbf{f}(\acute{z}, \mathbf{X}) = \mathbf{A}\mathbf{X} + \mathbf{S}$ in the inhomogeneous equation, and $\mathbf{f}(\acute{z}, \mathbf{X}) = \mathbf{A}\mathbf{X}$ in the homogeneous equation. Several different ODE integration methods have been developed; however, only two have come to dominate the modern market: the Runge-Kutta method and the Bulirsch-Stoer method. A third class of methods, called predictor-corrector, is still in use, but declining in popularity, and will not be discussed here. Press, Teukolsky, Vetterling and Flannery (1992) offer a good review of all these methods.

Runge-Kutta method

The Runge-Kutta method has its roots in the very simple (and inaccurate) Euler method, whose underlying idea is to rewrite the derivative in the ODE (110) as a discrete difference:

$$\frac{\Delta \mathbf{X}}{d} \approx \mathbf{f}(\acute{z}, \mathbf{X}) \tag{111}$$

where $d = \Delta\acute{z} = \acute{z}_{n+1} - \acute{z}_n$ is a discrete increment in $\acute{z}$, $\Delta\mathbf{X} = \mathbf{X}_{n+1} - \mathbf{X}_n$ is the corresponding increment in $\mathbf{X}$, and n is the step number. Say we are at step n and wish to advance to step $n+1$; we then only have information enough to calculate the function $\mathbf{f}$ at the beginning of that interval, and so we approximate $\mathbf{f}(\acute{z}, \mathbf{X}) \approx \mathbf{f}(\acute{z}_n, \mathbf{X}_n)$. Substituting these into Equation (111) and rearranging, we get the computational equation for a single Euler step:

$$\mathbf{X}_{n+1} = \mathbf{X}_n + \mathbf{f}(\acute{z}_n, \mathbf{X}_n)\, d \tag{112}$$

By starting at $\acute{z} = 0$, applying Equation (112) repeatedly until $\acute{z} = \acute{z}_{\text{top}}$ is reached, and saving $\mathbf{X}_n(\acute{z}_n)$ at every step, one obtains a numerical solution for $\mathbf{X}(\acute{z})$.

Order of accuracy In the limit $d \to 0$, we retrieve the original ODE (110), though this is possible in theory only. In practice, as d is reduced and the number of steps increases, the numerical solution becomes more and more representative of the underlying ODE — until it starts getting worse, due to the accumulated computational errors that inevitably accompany each step.

It is therefore crucial that too small a d not be needed for each step to be accurate. A power series expansion shows, however, that the leading error term of an Euler step is $\mathcal{O}\left(d^2\right)$, only one power of d smaller than the order of the increment $\Delta\mathbf{X}$. The Euler method is therefore only first-order accurate, and has little practical use on its own.

It can, however, be juggled with in order to improve accuracy. In the midpoint method, one takes a tentative Euler step to the midpoint of the interval (i.e. to $\acute{z}_{n+\frac{1}{2}} = \acute{z}_n + \frac{1}{2}d$), and then uses the $\mathbf{f}(\acute{z}_{n+\frac{1}{2}}, \mathbf{X}_{n+\frac{1}{2}})$ calculated at that point to advance across the entire interval d, replacing the $\mathbf{f}(\acute{z}_n, \mathbf{X}_n)$ in Equation (112). This makes the midpoint method second-order accurate.

The Runge-Kutta method is an extension of this same idea: three tentative steps are taken — two trial midpoints, and one trial endpoint — and fourth-order accuracy is attained. In our code, we used a similar scheme, the fifth-order Cash-Karp Runge-Kutta method.

Bulirsch-Stoer method

The Bulirsch-Stoer method is a "high-level" algorithm, which applies Richardson extrapolation to a lower-level algorithm, usually the modified midpoint method. This method is designed to advance $\mathbf{X}$ from location $\acute{z}$ by a large step D to destination $\acute{z} + D$, via a certain number n of smaller substeps, each of which is a midpoint-method step of length $d = D/n$ (the first and last substeps are slightly different).

Richardson extrapolation is then applied: a certain number m of modified-midpoint-method trials are performed, varying n each time, but (usually) keeping it small such that the substep size d is much larger than would otherwise be required for a solution with the desired accuracy. The m trial results at $\acute{z} + D$ are then fitted to a function of d, usually a polynomial or a rational function, which is then extrapolated to $d = 0$. One thus obtain an estimate of what the numerical solution $\mathbf{X}(\acute{z} + D)$ would have been, had we taken $n \to \infty$ substeps on a computer with infinite precision, yet this only cost a few trials with small values of n.

This procedure is repeated until the entire domain has been crossed in large steps. At each step, D and m are controlled by error-estimation routines.

Our choice

We chose the Runge-Kutta method for two main reasons. First, the Bulirsch-Stoer method is usually both more efficient and more accurate than Runge-Kutta when the derivatives function $\mathbf{f}(\acute{z}_n, \mathbf{X}_n)$ is continuous and well-behaved. However, it can be the source of stability problems when $\mathbf{f}(\acute{z}_n, \mathbf{X}_n)$ is non-smooth, for example when it is interpolated from a look-up table; in this case, the Runge-Kutta method is superior. In our current flow model, the $\mathbf{A}$ and $\mathbf{S}$ matrices indeed consist of smooth, analytical functions, but in future applications, we will likely calculate a numerical zero-order solution and store it in look-up tables instead of using the analytical expressions in Equations 72 to 76 (page 48). In preparation for this, we found it preferable to implement the Runge-Kutta method.

Naysayers will point out that a numerical zero-order solution could be fitted to analytical functions, which could then be substituted in place of Equations 72 to 76, with the added benefit of reducing computation time. But this involves the extra work of finding good fitting functions, a task not as easily automated as the generation of look-up tables.

Our second reason for preferring the Runge-Kutta method is that our numerical integration task is about the journey, not (just) the destination, so to speak. The Bulirsch-Stoer method is very efficient at getting from start to finish, swiftly striding across the domain in comparatively large steps of size D. While the solution at every step may be very accurate, however, certain solution features may not necessarily be resolved, precisely due to the large step size (Figure 8). Yet we are interested in the behaviour of the solution throughout the domain, not only at the end point. With its explicitly stated step size, we felt that the Runge-Kutta method provided us with greater control over the mid-domain resolution.

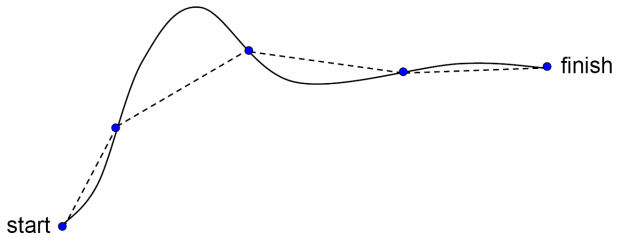


Figure 8: The Bulirsch-Stoer method: the steps (dots) are very accurate, but they constitute a low-resolution representation (dotted line) of the true solution (full line) throughout the domain.

Step size control A related issue was adaptive step size control in the Runge-Kutta method. Ideally, one should exert some kind of automatic, adaptive step size control, based on an error-estimation criterion, such that small steps are taken where $\mathbf{f}(\acute{z}_n, \mathbf{X}_n)$ changes rapidly, and larger steps where it is flat.

However, our task implies adding several shots at the same specific positions (values of $\acute{z}$). Adaptive step size control would have produced shots saved at different values of $\acute{z}$, and would have required interpolation.

We chose to control step size manually, imposing small steps near the surface, where most of the action happens, and large steps near the top of the domain, where the perturbation slowly tapers off to zero.

5.5 Analysis of the equation

Equation in the far field

In the far field ($k\acute{z} \to \infty$), the source term $\mathbf{S}$ rapidly falls to zero as $e^{-k\acute{z}}$, and the matrix $\mathbf{A}$ tends toward a relatively simple form $\mathbf{A}_\infty$, independent of $\acute{z}$. We are thus left with the equation

$$\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{A}_\infty \mathbf{X} \tag{113}$$

where

$$\mathbf{A}_\infty = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ -k^4 & 0 & 2k^2 & 0 & 0 & 0 & -\frac{2ik^3}{U_{*0}^2} & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & k^2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & k^2 & 0 \end{pmatrix}$$

This matrix can be brought to a matrix $\mathbf{G}$ in Jordan's canonical form by a rotation matrix $\mathbf{T}$ (also independent of $\acute{z}$) such that $\mathbf{T}^{-1}\mathbf{A}_\infty\mathbf{T} = \mathbf{G}$ and, conversely, $\mathbf{A}_\infty = \mathbf{T}\mathbf{G}\mathbf{T}^{-1}$, where

$$\mathbf{T} = \begin{pmatrix} 0 & -k^{-3} & -3k^{-4} & -6k^{-5} & 0 & k^{-3} & -3k^{-4} & 6k^{-5} \\ 0 & k^{-2} & 2k^{-3} & 3k^{-4} & 0 & k^{-2} & 2k^{-3} & 3k^{-4} \\ 0 & -k^{-1} & -k^{-2} & -k^{-3} & 0 & k^{-1} & -k^{-2} & k^{-3} \\ 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ -k^{-1} & 0 & 0 & 0 & k^{-1} & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2iU_{*0}^2k^{-4} & 0 & 0 & 0 & 2iU_{*0}^2k^{-4} \\ 0 & 0 & 0 & 2iU_{*0}^2k^{-3} & 0 & 0 & 0 & 2iU_{*0}^2k^{-3} \end{pmatrix} \tag{114}$$

$$\mathbf{T}^{-1} = \begin{pmatrix} 0 & 0 & 0 & 0 & -\frac{k}{2} & \frac{1}{2} & 0 & 0 \\ \frac{k^2}{4} & 0 & -\frac{3k}{4} & \frac{1}{2} & 0 & 0 & \frac{3ik^2}{8U_{*0}^2} & 0 \\ -\frac{k^4}{4} & \frac{k^3}{4} & \frac{k^2}{4} & -\frac{k}{4} & 0 & 0 & -\frac{5ik^3}{8U_{*0}^2} & \frac{3ik^2}{8U_{*0}^2} \\ 0 & 0 & 0 & 0 & 0 & 0 & \frac{ik^4}{4U_{*0}^2} & -\frac{ik^2}{4U_{*0}^2} \\ 0 & 0 & 0 & 0 & \frac{k}{2} & \frac{1}{2} & 0 & 0 \\ -\frac{k^2}{4} & 0 & \frac{3k}{4} & \frac{1}{2} & 0 & 0 & -\frac{3ik^2}{8U_{*0}^2} & 0 \\ -\frac{k^4}{4} & -\frac{k^3}{4} & \frac{k^2}{4} & \frac{k}{4} & 0 & 0 & -\frac{5ik^3}{8U_{*0}^2} & -\frac{3ik^2}{8U_{*0}^2} \\ 0 & 0 & 0 & 0 & 0 & 0 & -\frac{ik^4}{4U_{*0}^2} & -\frac{ik^2}{4U_{*0}^2} \end{pmatrix}$$

and

$$\mathbf{G} = \begin{pmatrix} -k & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -k & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -k & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -k & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & k & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & k & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & k & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & k \end{pmatrix}$$

Multiplying both sides of Equation (113) by $\mathbf{T}$, we have

$$\mathbf{T}\frac{\partial \mathbf{X}}{\partial \acute{z}} = \mathbf{T}\mathbf{A}_\infty\mathbf{X} = \mathbf{T}\mathbf{A}_\infty\mathbf{T}^{-1}\mathbf{T}\mathbf{X}$$

or simply

$$\frac{\partial \mathbf{R}}{\partial \acute{z}} = \mathbf{G}\mathbf{R} \tag{115}$$

where $\mathbf{R} = \mathbf{T}\mathbf{X}$. It is readily shown that the solution to this equation is of the form

$$\mathbf{R} = \begin{pmatrix} a_1 \\ a_2 + a_3\acute{z} + a_4\acute{z}^2 \\ a_3 + a_4\acute{z} \\ a_4 \\ 0 \\ 0 \\ 0 \\ 0 \end{pmatrix} e^{-k\acute{z}} + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 0 \\ a_5 \\ a_6 + a_7\acute{z} + a_8\acute{z}^2 \\ a_7 + a_8\acute{z} \\ a_8 \end{pmatrix} e^{k\acute{z}} \tag{116}$$

where $a_1, \ldots, a_8$ are polynomial coefficients determined by the boundary conditions.

Exponentially growing and decaying solutions

In the far field, the exponential factors dominate the polynomials, and we thus have four exponentially decaying and four exponentially growing solution elements. Now each element of $\mathbf{X}$ is a linear combination of the elements of $\mathbf{R}$ through $\mathbf{X} = \mathbf{T}^{-1}\mathbf{R}$. So if $\mathbf{R}$ contains exponentially growing solution elements, then the first-order perturbation variables grow exponentially with height in the far field. This is of course a physically unrealistic solution: we assume that the flow perturbation caused by the orography dies out with height. We are therefore only interested in the solutions that decay exponentially with height.

This analysis, in which the exponentially growing and decaying solutions are readily segregated, is only possible in the far field where $\mathbf{A} \to \mathbf{A}_\infty$. In the near field, $\mathbf{A}$ cannot be transformed into a Jordan matrix independently of $\acute{z}$. Rather, locally increasing solutions may change character and become decreasing as one moves higher in $\acute{z}$, and vice-versa.

Equation (103) must therefore be solved as is. This means that each individual "shot" $\mathbf{X}_i$ is a linear combination of exponentially growing and decaying solutions — and the growing solutions always end up dominating as $\acute{z}$ increases. Even if we found start conditions for numerical integration that would, on paper, only bring out the exponentially decaying solutions, tiny numerical errors due our finite machine precision would in fact also excite the exponentially growing solutions.

This fact poses a dilemma on how high our upper boundary should be. As described in Section 2.7, we wish $k\acute{z}_{\text{top}}$ to be high enough for the coordinates to become Cartesian again and for the perturbation to become negligible. However, if $k\acute{z}_{\text{top}}$ is too large, the undesirable, exponentially growing solutions end up swamping the interesting, exponentially decaying solutions at the upper boundary.

Only the decaying solutions can satisfy the upper boundary conditions, but extracting them from the much larger growing solutions becomes numerically difficult: it is an extension of the familiar problem of calculating the small difference between two very large numbers, when only a limited number of decimal places are available. In mathematical terms, the matrices $(\mathbf{Q}\mathbf{X}^{\text{top}}_{\text{hom}})$ of Equation (109) and $\begin{pmatrix} -\mathbf{X}^{\text{bu}}_{\text{hom}} & \mathbf{X}^{\text{td}}_{\text{hom}} \end{pmatrix}$ of Equation (109) become ill-conditioned, such that the solution for $\mathbf{\Phi}$ is rife with numerical error due to finite machine precision.

The $kz_0 \to 0$ problem

This problem is accentuated as $kz_0 \to 0$, i.e. as the domain becomes large and the terrain, smooth. With our upper boundary at $k\acute{z}_{\text{top}} = 10$, we found that the

problem becomes numerically ill-posed when kz_0 falls below 5×10^{-6}. This was the best result we could obtain, after optimizing the position of our fitting point at $k\acute{z}_{\text{fit}} = 0.95$ (see "Shooting to a fitting point" on page 65).

An example of the resulting numerical noise affecting $\mathcal{W}$ when $kz_0 = 6.28 \times 10^{-8}$ is shown on Figure 9. The problem is identical for the other variables: the noise becomes too intense near the fitting point for any meaningful matching to occur.

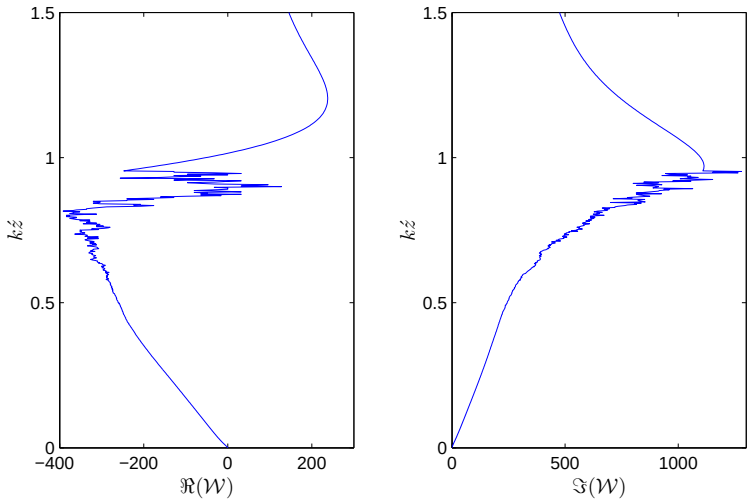


Figure 9: An example of the numerical noise on the real and imaginary parts of $\mathcal{W}$ for $kz_0 = 6.28 \times 10^{-8}$.

We can therefore not cover the entire range of non-dimensional roughnesses defined in Table 2 (page 62): we would like to reach down to $kz_0 = 6.28 \times 10^{-8}$, but approximately two decades are missing. This illustrated on Figure 10: the calculated value of the variable $\mathcal{W}$ at a non-dimensional height of $k\acute{z} = 0.9$ is plotted over the desired range of kz_0. This non-dimensional height is in the most affected region, namely just below the fitting point $k\acute{z}_{\text{fit}} = 0.95$. The absolute value of $\mathcal{W}$ clearly shifts to an unstable regime around $kz_0 = 10^{-6}$, but when we separate $\mathcal{W}$ into its real and imaginary parts, it is apparent that the real part is even more sensitive: it becomes unstable around $kz_0 = 5 \times 10^{-6}$. This is of particular concern, since the real part is responsible for the phase shift with respect to the terrain[2].

Despite our best efforts at optimizing various parameters of the numerical integration (step size, position of the fitting point, starting values for individual "shots"), we have not yet been able to reduce the $kz_0 \to 0$ problem to a more acceptable level. Potential ways around this problem will be discussed in Section 7.2.

5.6 Analytical solution

In the course of performing manipulations with Mathematica, we fortuitously found an analytical solution $\mathbf{X}_{\text{a}}(\acute{z})$ to the inhomogeneous equation (103). This solution, detailed in Appendix C, decays exponentially with height, and thus satisfies the upper boundary conditions; but it does not, unfortunately, even come close to satisfying the lower boundary conditions: the perturbation grows enormously near the surface.

[2] $\Re\{\mathcal{W}\} = 0$ would imply symmetric flow given a symmetric hill. By "symmetric flow" we mean that the streamlines would be perfectly symmetric with respect to the hill's center. $\mathcal{W}$ itself would of course be perfectly *anti*symmetric: the air flows up one side of the hill and down the other. This is why the real and not the imaginary part of $\mathcal{W}$ is responsible for the phase shift.

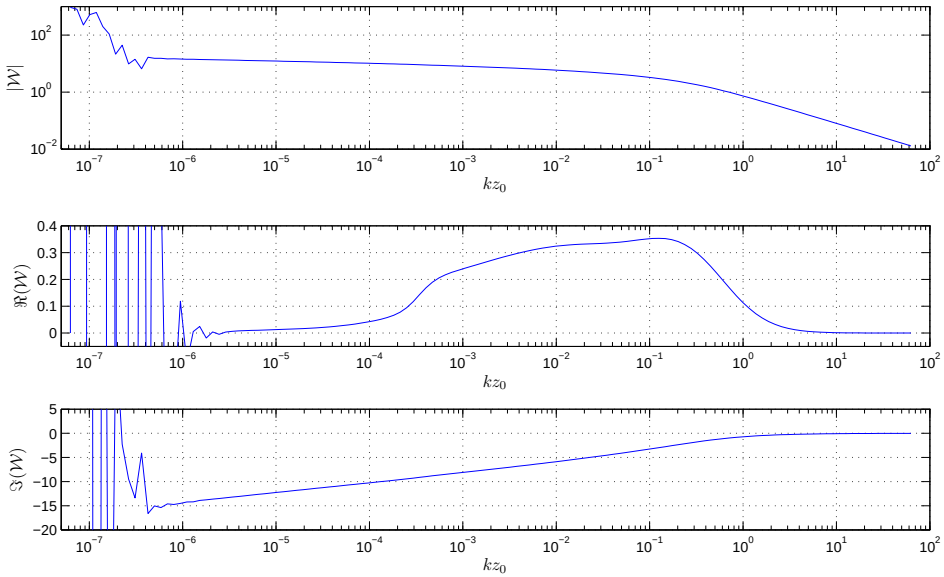


Figure 10: The $kz_0 \to 0$ problem, illustrated for (from top to bottom) the absolute value, real part, and imaginary part of the variable $\mathcal{W}$ at a non-dimensional height of $k\acute{z} = 0.9$.

Repairing the analytical solution would require four shots with the homogeneous equation, and large weighting coefficients $\mathbf{\Phi}$, thus enhancing the numerical noise with which they are contaminated. The analytical solution can therefore not be put to practical use in any direct way, though a possible roundabout way of using it will be discussed in Section 7.2 (page 100).

5.7 Processing the terrain

Making the terrain periodic

Our spectral treatment of the first-order equations requires that the calculation domain be interpreted as horizontally periodic. This is not a good approximation to reality if the terrain of interest is left unprocessed (Figure 11), for two reasons:

1. In the periodic interpretation, the lee of one replica of the hill affects the inflow over the next replica, since it is so close by.
2. At the interface between two adjacent replicas, there is an artificial, vertical "cliff" that may lead to spurious flow disturbances.

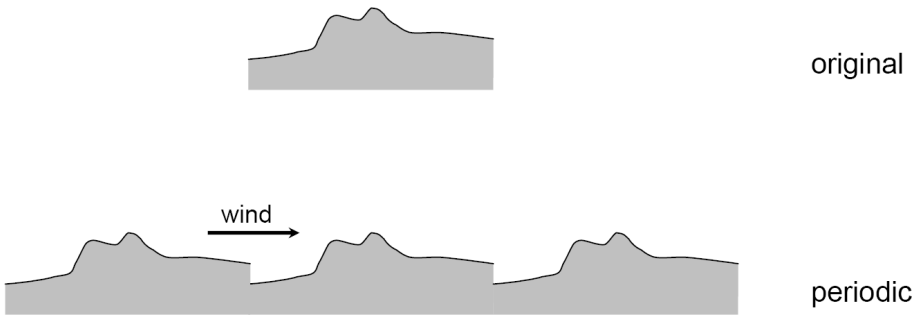


Figure 11: A naive periodicizing of the terrain.

Buffering A judicious buffering of the terrain is our answer to problem 1: a fictitious flat plateau of length L_{buf} is appended to each end of the original terrain, such that successive replicas of the hill are far away enough from each other for their flows not to interact (Figure 12). L_{buf} must not, however, be so large as to unduly increase the size of the calculation domain (and thereby the computational load). We choose an L_{buf} that is just large enough such that there is no discernible change in the flow field over the region of interest when doubling L_{buf}. In the cases where the ends of the original terrain are not quite horizontal, a short extra buffer is added to smoothly flatten out the ends before appending the flat plateaus, so as to avoid creating sharp edges.

Mirroring Problem 2 is addressed by mirroring the terrain (Figure 12). This ensures that both ends of the terrain are at the same height, thus avoiding the creation of artificial cliffs. In the process of mirroring, the calculation domain is doubled in size. This does not, however, translate into a doubling of the computational cost, since the forward and backward flows (i.e. the flow fields generated by wind coming from two opposite directions) are calculated simultaneously.

Alternatively, one could have chosen to match the ends of the terrain by a long, gentle-sloped buffer. This would have increased the computational cost without providing the benefit of calculating the forward and backward flows at once.

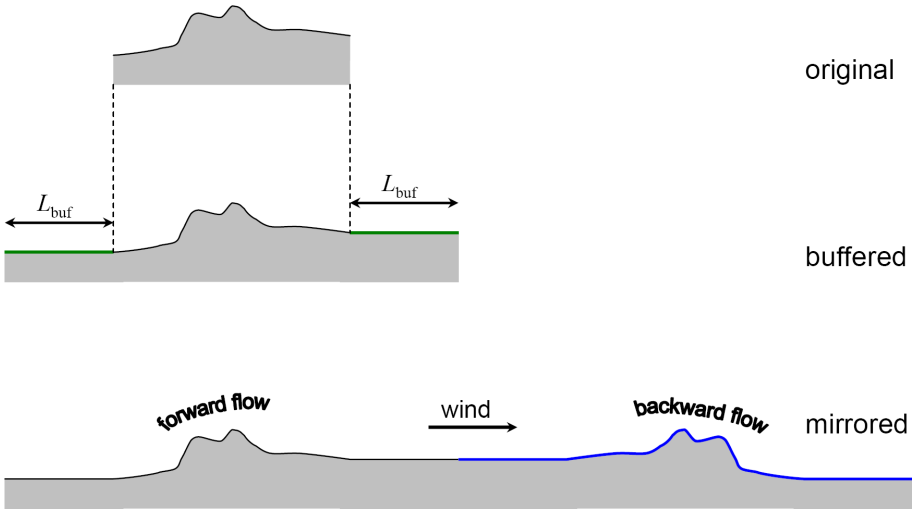


Figure 12: Buffering and mirroring of the terrain.

The mirroring tactic can be extended to the 3D case: the buffered terrain, placed in one quadrant, is mirrored into the three other quadrants. This quadruples the domain size, but also the information provided by a single calculation, namely the forward and backward flows for two different wind directions relative to the hill, as illustrated in Figure 13. An exception is when the wind direction is exactly parallel/perpendicular to the mirroring axes, in which case only one unique forward-backward flow pair is yielded; this configuration should therefore be avoided.

Smoothing filter

Given that the terrain is sampled at a finite resolution, our spectral representation of the terrain poses a limitation to how sharp terrain features may be. Sharp edges cannot be faithfully represented by anything less than an infinite number

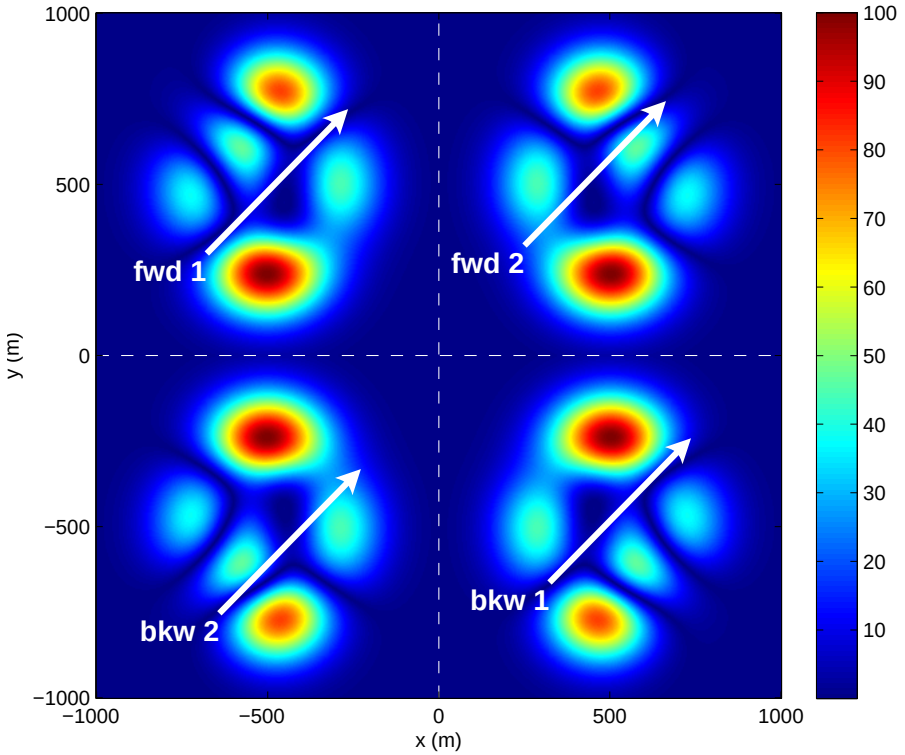


Figure 13: Mirroring a hypothetical 3D terrain. Terrain height (m) is indicated by the colour scale. Two different forward flows ('fwd 1' and 'fwd 2') and their corresponding backward flows ('bkw 1' and 'bkw 2') are calculated at once. The dotted lines represent the mirroring axes.

of wavenumbers; cutting off the spectrum at a finite wavenumber, as is necessary for numerical work, results in ringing artifacts. Even with an infinite number of wavenumbers, the representation is not adequate for our purposes, most notably for sharp vertical cliffs: as the number of wavenumber rises, the overshoot at the top of the cliff is reduced in width, but converges to a finite height — 9% of the cliff height, in the case of a square wave (Figure 14). This is known as the Gibbs phenomenon. The overshoot is of great concern to us, for no matter how narrow it is, it still acts in effect as a wind-blocking wall, leading to spurious effects on the flow, notably a recirculation region in its lee.

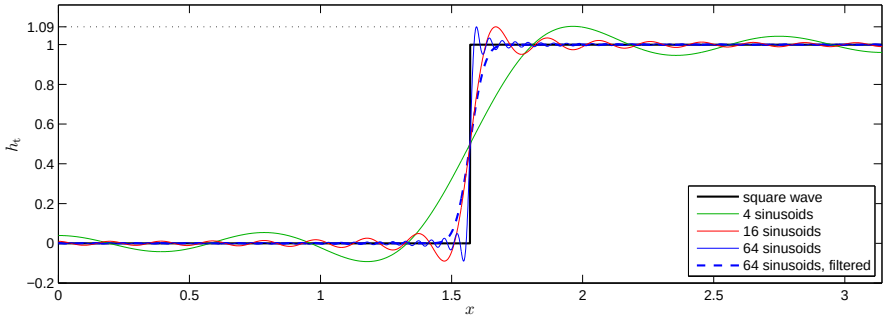


Figure 14: The Gibbs phenomenon: a square wave approximated by a sum of sinusoids. The overshoot can be eliminated by applying a Gaussian filter, at the cost of blurring the sharp edges and cliffs.

We circumvent this problem by first sampling the terrain at a high resolution, and then applying a smoothing filter — at the cost of blurring the terrain somewhat. The dampened high wavenumbers can then be discarded, and the filtered terrain re-sampled at a lower resolution, without introducing ringing artifacts. Our filter was a Gaussian, with a standard deviation σ chosen so that no significant terrain feature was unduly blurred out. Vertical cliffs and sharp edges, however, inevitably become smoothed out; this limitation, intrinsic to the spectral nature of our flow model, must be kept in mind.

An example of the effects of filtering on the power spectrum of a terrain (Askervein hill, discussed in Section 6.4) is shown on Figure 15, and the corresponding unfiltered and filtered height profiles are compared on Figure 16. Filtering results in changes of local height of at most 1 m, which is unlikely to affect the flow pattern, and is in any case less than the resolution of the height contour map from which the terrain was digitized.

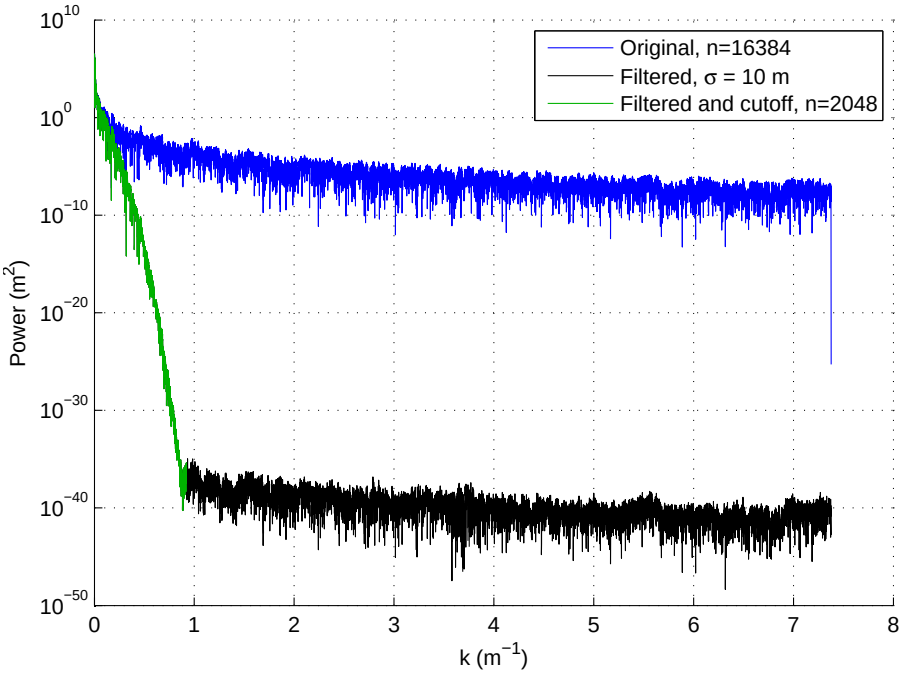


Figure 15: Power spectrum of Askervein hill line A, before (blue) and after (green/black) applying a smoothing filter. After filtering, the wavenumbers in black are discarded as they are negligible.

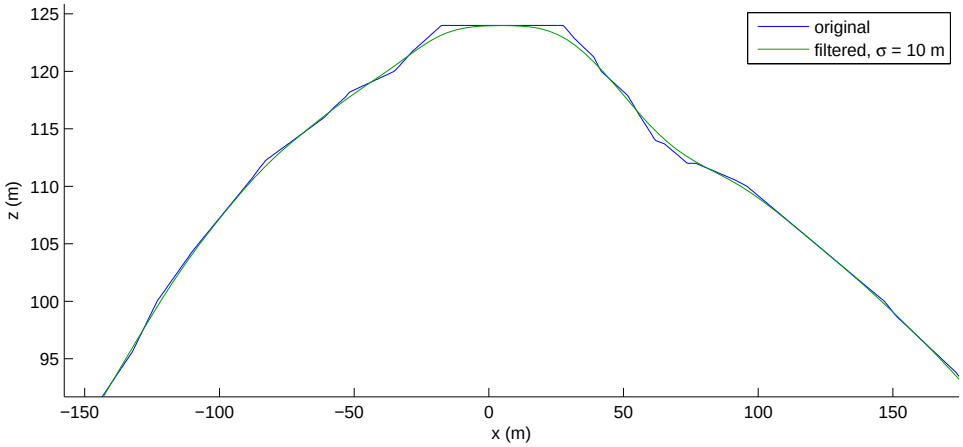


Figure 16: Height profile of Askervein's hilltop along line A, before and after applying a smoothing filter.

6 Results

We now present a comparison of RAMSIM calculations with experimental data as well as calculations by other flow models for a number of test cases of flow over orography: two wind-tunnel experiments with idealized terrains, and two field experiments over real terrain. These two types of experiment are each associated with different advantages and difficulties, mainly due to their difference in scale.

In wind tunnels, one is free to build all forms of idealized hills or replicas of real terrain at miniaturized scales of $1 : 10^3$ or so, and to reproduce experiments at will. However, comparatively large roughness elements are required to maintain aerodynamic roughness, leading to larger roughnesses than might otherwise be desired; one is usually limited to equivalent full-scale roughnesses of the order of $z_0 \approx 1$ m, typical of forests. Also, the sheer size of the roughness elements limits the lowest height at which meaningful measurements can be taken, since the flow in the lowest layer is contaminated by local 3D effects caused by individual roughness elements. The entire inner layer is most often disturbed in this way.

In measuring the miniaturized flow, one is constrained to instruments that have a very fine spatial resolution. Hot-wire anemometers, the most usual instrument for velocity measurements, have a rectifying action, making them insensitive to flow reversal. Pulsed-wire anemometers (Bradbury and Castro 1971) can detect the direction of flow, but have traditionally been more bulky, thus posing a limit to how close to a wall they can be used; ironically, this is precisely where flow reversal usually occurs. Recent advances in near-wall pulsed-wire anemometry have been made by Schober, Hancock and Siller (1998) and Hancock (2004). Laser Doppler anemometers, first developed in the early 1970s (see for example Adrian 1971 and McLaughlin 1973), are increasingly being used in wind tunnel experiments. In usual setups, they too are insensitive to flow reversal, as they detect the absolute value of the velocity component normal to a fringe pattern created at the intersection of two laser beams.

Field measurement campaigns seize on the opportunity of observing the full-scale flow over real terrain. Experiments are, by definition, not reproducible in a controlled fashion, as one is at the mercy of weather-related variability in wind speed, wind direction, and thermal stratification. In order to capture the spatial variation of the flow field over orography, one needs a great many expensive and delicate instruments (wind vanes, cup and sonic Doppler anemometers) perched on masts spread over a large area. The depth of the wake or recirculation region behind hills is of the same order as the hill height, which is usually much higher than the commonly-used 10-meter towers; higher towers are more expensive to deploy. Cost is thus a major limiting factor on spatial resolution, especially far from the ground. The advent of sodar and lidar remote-sensing technology (sonic- and laser-based equivalents of radar), reviewed by Zak (2003), promises to increase the availability of velocity measurements at heights beyond the reach of run-of-the-mill meteorological masts.

Here we draw on the strengths of both types of experiments: we compare our model results to wind-tunnel measurements over periodic sinusoidal hills (Section 6.1) and an isolated hill (Section 6.2), and to field measurements over Hjardemål escarpment (Section 6.3) and Askervein hill (Section 6.4). Using the look-up table built earlier, RAMSIM calculation time was at most a few seconds in all cases.

6.1 Sinusoidal hills in a wind tunnel

The simplest possible test case for RAMSIM is a periodic sinusoidal hill, i.e. a single wavenumber. No buffering, mirroring or low-pass filtering of the terrain

is required. Athanassiadou and Castro (2001) carried out measurements of the neutral flow over rough, sinusoidal hills in the wind tunnel facilities in the Environmental Flow Research Laboratory (EnFlo), University of Surrey. In this wind tunnel, series of twelve sinusoidal hills of wavelength $\lambda = 330$ mm were built from a wooden frame and a smooth plywood cover, onto which an aluminium mesh was spread to provide some roughness. Two such sets were built: "small hills" with amplitude $a = 10.5$ mm and "large hills" with $a = 21$ mm. The maximum slope was 0.2 for the small hills, and 0.4 for the large hills.

The roughness was determined to be $z_0 = 0.337$ mm by measuring a vertical velocity profile over a flat surface covered by the aluminium mesh. The shear stress was constant over the lower 16 % of the boundary layer, and then decreased with height. The boundary-layer depth over the flat surface, taken as the height where the shear stress is reduced to 5% of its surface value, was measured to be $h_{\mathrm{BL}} = 100$ mm. Assuming a typical ABL height of 500 m, this represents a model scale of 1:5000. The corresponding full-scale roughness is therefore 1.6 m, which is a typical value for forests or cities. This scaling must, however, be regarded as very coarse, since boundary-layer heights in the atmosphere and in a wind tunnel are determined by different factors. The trough-to-peak heights of the small and large hills were approximately 0.2 and 0.4 times the depth of the undisturbed boundary layer, respectively.

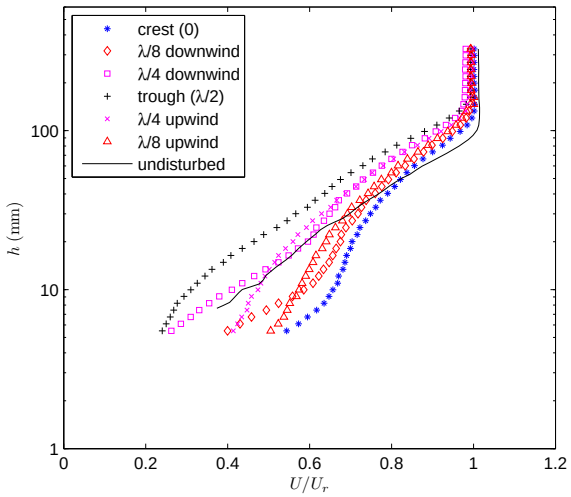
Athanassiadou and Castro measured the mean flow and turbulence statistics using single hot wire, cross wire, and pulsed wire probes. Surface pressure was measured using pressure tappings placed flush with the surface, in the middle of the diamond-shaped 'holes' of the aluminium mesh — this was where the difference between the static pressure and that measured in the free stream was minimum, and where the measurements were most repeatable. All measurements were carried out over the 10th and 11th hills (counting from the inlet of the wind tunnel), at which point the flow had reached an approximate equilibrium.

Small hills

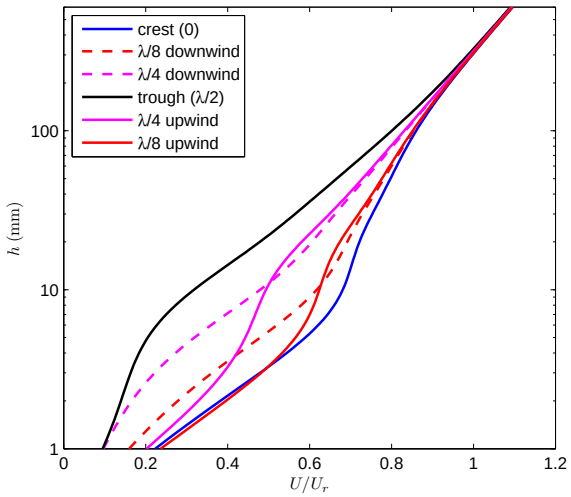
Figure 17a shows vertical profiles of the mean velocity measured by Athanassiadou and Castro using a cross-wire probe at various locations over the 10th hill: the crest ($x = 0$), the trough between the 10th and 11th hills ($x = \lambda/2$), as well as two locations on the upwind ($x = -\lambda/4, -\lambda/8$) and downwind ($x = \lambda/4, \lambda/8$) slopes. Equivalent profiles were calculated with RAMSIM, assuming a periodic sinusoidal hill (Figure 17b). For verification, we also performed calculations over a terrain consisting of a series of twelve sinusoidal hills (as in the wind tunnel), then buffered and mirrored in the usual way. This caused only negligible differences on the wind speed profiles, compared to the simply periodic sinusoidal hill.

The normalization velocity U_r for the measured profile was the free-stream velocity, i.e. the uniform velocity measured at heights above ground level greater than $h = 200$ mm. RAMSIM assumes the undisturbed profile to be logarithmic, never levelling off to a uniform free-stream velocity. For the calculations, U_r was therefore arbitrarily chosen so that the calculated profile above the crest matches the measured profile at a height of 30 mm; this appeared to be the location where the spurious effects affecting the laboratory and computed flows would be of least importance.

There is a qualitative agreement between the shapes of the measured and calculated profiles, with curves crossing each other in the same order in both cases. The flow remains attached throughout. Quantitative comparison is difficult due to the particularities of the flow inside the wind tunnel. RAMSIM assumes the undisturbed flow to be logarithmic, i.e. with a uniform $\overline{u_1' u_3'}$. The measured undisturbed flow, however, has a uniform $\overline{u_1' u_3'}$ only in the lower 20 mm or so. Above



(a) cross-wire measurements



(b) RAMSIM calculation

Figure 17: Vertical profiles of the normalized velocity over the small hills. Also shown on (a) is the undisturbed profile over a flat surface with the same roughness.

this height, $\overline{u_1' u_3'}$ gradually decreases in magnitude, so that at a given height, more momentum is transported in from above than out to lower layers, leading to supra-logarithmic flow up to the top of the boundary layer at a height of about 100 mm. Beyond this, $\overline{u_1' u_3'} = 0$, and the velocity profile flattens out. Moreover, the lower ends of the measured profiles may be affected by undetermined flow features created by the individual roughness elements, which are 3.46 mm high.

RAMSIM results were also compared with the available surface pressure measurements (Figure 18; note that the hill crest is centered at $x = 0$). There is some degree of arbitrariness associated to the magnitude of the calculated profile, due to the choice of matching point for the normalization velocity U_r. Moreover, the pressure measurements are taken within the aluminium mesh, so that they may very well be affected by flow features created by individual roughness elements. These factors must be kept in mind when quantitatively comparing the measured and calculated profiles point by point. There is nevertheless excellent agreement with the measurements on the upwind slope, the hill crest, and even half of the downwind slope. In the trough, the pressure perturbation is overestimated. The calculated pressure profile is slightly more de-phased with respect to the hill shape:

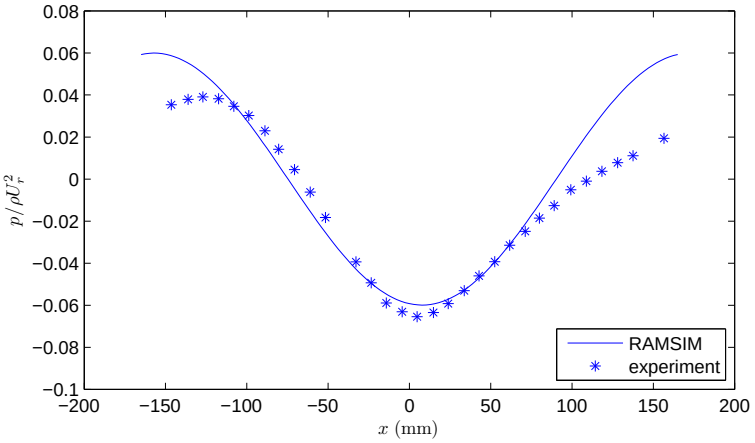


Figure 18: Surface pressure (normalized by ρU_r^2) over the small hills. The hill crest is centered at $x = 0$.

the pressure minimum occurs 7.8 mm behind the hill crest, whereas the measured minimum is around $x = 5$ mm. The measured pressure reduction at the crest is greater in magnitude than the pressure increase in the trough. This is not captured by RAMSIM, which is inherently limited to a zero-mean sinusoidal perturbation in the case of a sinusoidal hill.

Large hills

With their maximum slope of 0.4, the large hills are steep enough to cause flow separation. This is nicely illustrated by a plot of the calculated streamlines, on which the recirculation bubble is clearly visible (Figure 19). The extent of this recirculation region is perhaps best characterized by the locus of $u = 0$, i.e. where the mean flow is purely vertical. This lies below the separating (and, for this 2D flow, reattaching) streamline, but is much easier to measure. Using a pulsed wire anemometer, Athanassiadou and Castro located five points of this curve (Figure 20). RAMSIM successfully models the presence of a recirculation region, but underestimates its vertical extent by about half, and also, to a lesser degree, its horizontal extent.

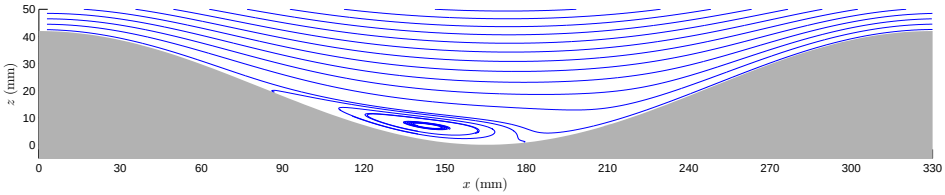


Figure 19: Streamline plot of the flow over the large hills, as calculated by RAMSIM.

Figure 21 shows the measured and calculated vertical profiles of the mean velocity over the large hills, taken at the same horizontal positions as previously over the small hills. Compared to the small hills' case, there is an additional obstacle to quantitative comparison between the calculated and measured profiles: cross-wire probes have a rectifying action, so that measurements in a recirculation region do not show the direction of flow, whereas the calculations clearly show flow reversal near the surface in the trough.

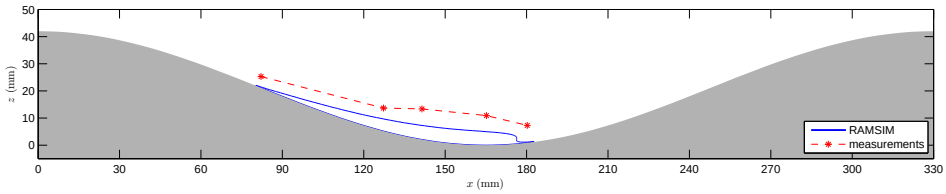
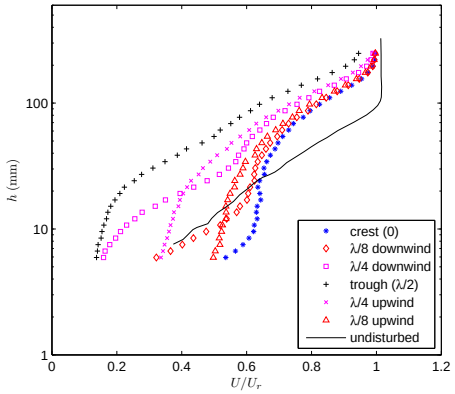
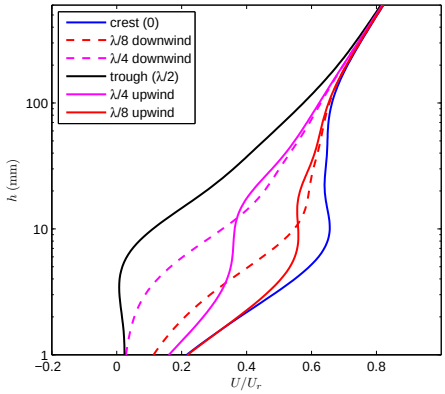


Figure 20: Locus of $u = 0$: RAMSIM calculation compared to pulsed wire measurements by Athanassiadou and Castro (2001).



(a) cross-wire measurements



(b) RAMSIM calculation

Figure 21: Vertical profiles of the normalized velocity over the large hills. Shown again on (a) is the undisturbed profile over a flat surface with the same roughness.

The comparison between the measured and calculated surface pressure profiles highlight the greater influence of non-linear effects in the large hills' case (Figure 22). The shape of the measured pressure perturbation deviates markedly from the sinusoidal to which RAMSIM is constrained. The measured pressure maximum ($x = -100$ mm) is likely associated to the point where the flow reattaches. The measured pressure minimum occurs closer to the crest (near $x = 3$ mm) than in the small hills' case; this is not captured by RAMSIM for which the perturbation is simply scaled up in magnitude from the small hills' case. The measured pressure perturbation is again greater in magnitude at the crest than in the trough. RAMSIM apparently underestimates the former and calculates the latter correctly (but not the location of the pressure maximum); though again, the calculated magnitudes depend on our choice of normalization U_r.

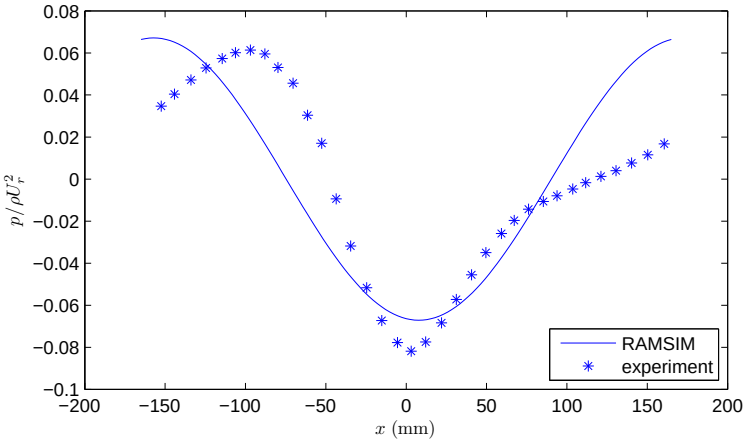


Figure 22: Surface pressure (normalized by ρU_r^2) over the large hills. The hill crest is centered at $x = 0$.

Pressure drag

The pressure drag per unit wavelength (normalized by ρU_r^2) is related to the surface pressure p and the slope of the terrain by

$$F_p = \frac{1}{\lambda} \int_{-\lambda/2}^{\lambda/2} \frac{p}{\rho U_r^2} \frac{\partial h_t}{\partial x} dx \tag{117}$$

where h_t is the surface elevation and λ is the wavelength of the hill. The slope of the sinusoidal hill is simply given by $\partial h_t/\partial x = (2\pi a/\lambda)\sin(2\pi x/\lambda)$. The measured and calculated curves $p(x)$ of Figures 18 and 22 were fit to piecewise third-order polynomials[3], which were then used to integrate Equation (117).

The calculated pressure drag was too small by a factor of 2 for the small hills, and 3 for the large hills (Table 3). This is mainly a consequence of the overestimation of the pressure in the wake region, and in the large hill's case, a misplaced flow reattachment point, leading to an underestimation of the pressure on the upwind slope.

Table 3: Pressure drag over the small and large sinusoidal hills

	$F_p \times 10^3$	
	small hills	large hills
experiment[4]	1.73	6.35
RAMSIM	0.89	1.99
discrepancy	-49%	-69%

Athanassiadou and Castro (2001) compare their measured drag forces to results by Newley (1985) and Wood and Mason (1993). Newley performed numerical simulations over hills of low slope (<0.25), and found that the pressure drag per unit wavelength in this case could be parameterized as $F_p = A_p s_{\max}^2 (U_{*0}/U_r)^2$

[3] The curves were periodicized before fitting, to ensure proper behaviour at the ends of the curves.

[4] The values cited by Athanassiadou and Castro (2001) are slightly different, probably because they used a different fitting function.

where $s_{\max}$ is the maximum slope. The value of the constant A_p depended strongly on the turbulence closure used, with $A_p \approx 12$ for a one-equation mixing-length closure, and $A_p \approx 6$ for a second-order stress-transport closure. Wood and Mason later found a dependency of the pressure drag on λ/z_0 (or, in other words, our non-dimensional roughness kz_0), and that the discrepancy between the results given by the two turbulence closures decreases as non-dimensional roughness increases; they derived heuristic formulas accounting for this effect as well as the effect of steeper slopes. Their predictions for the constant A_p for our case of $\lambda/z_0 = 10^3$ are compared with Athanassiadou and Castro's experimental results and RAMSIM's calculations in Table 4.

RAMSIM's A_p is by definition invariant under change of scaling velocity U_{*0} and hill amplitude a, since it is the quotient of $F_p \propto p\,\partial h_\mathrm{t}/\partial x \propto (aU_{*0}^2)a$ and $s_{\max}^2(U_{*0}/U_r)^2 \propto a^2U_{*0}^2$. It is smaller by 30% to 50% relative to the other values listed. Oddly, the discrepancy is roughly the same (even a bit smaller) in the large hills' case as in the small hills' case, whereas the discrepancy in F_p was larger in the large hills' case. This may indicate that our choice of normalizing velocity $U_r \propto U_{*0}$ was too high in the large hills' case, due to placing the matching point in a region where RAMSIM exaggerates the speed-up, the result being that the calculated pressure drag ($\propto U_{*0}^2$) in Table 3 was excessively deflated.

Table 4: Measured and predicted values of the pressure-drag constant A_p

	A_p
Athanassiadou and Castro (2001) small hills	12.33
Athanassiadou and Castro (2001) large hills	10.9
RAMSIM calculation (small and large hills)	6.53
Wood and Mason (1993) mixing-length closure	12.7
Wood and Mason (1993) second-order closure	9.52

6.2 Isolated hill in a wind tunnel

Not as simple as single-wavenumber periodic hills, but perhaps more commonly found in nature, is the case of a single, isolated hill. Ross, Arnold, Vosper, Mobbs, Dixon and Robins (2004) carried out, also at EnFlo, a set of wind-tunnel experiments over single two-dimensional hills with a height profile given by $h_\mathrm{t} = H\cos^2(\pi x/L)$ for $-L/2 \le x \le L/2$, amid otherwise flat terrain. They also performed numerical simulations of the same flows using a range of different turbulence closures.

Experimental setup

Two different hills were studied: both were of height $H = 0.229$ m, but the "shallow" hill had $L = 2.4$ m, while the "steep" hill had $L = 1.0$ m. With maximum slopes of 0.3 and 0.72, respectively, these hills are steeper than those studied by Athanassiadou and Castro (2001).

Roughness elements 20 mm high and 80 mm wide were placed over the entire surface, with a streamwise spacing of 1 m and a lateral spacing of 1.6 m. The resulting roughness length was determined to be $z_0 = 2.3$ mm from a log-linear fit

to the measured vertical velocity profile over the flat surface. The boundary-layer depth was approximately 1 m.

Assuming a model scale of 1:1000, the corresponding full-scale roughness would be 2.3 m, representative of tall forests or cities, and the full-scale hill would be 230 m in height.

Ross et al. were able to produce two sets of thermally stratified conditions in the wind tunnel through a combination of heating the inlet air using multi-level heaters and cooling the surface using a system of cool-water pipes under the floor panels. However, only the experiments with neutrally stratified conditions will be considered here, since RAMSIM is currently limited to thermally neutral flows.

Velocity was measured with a two-component laser Doppler anemometer, as well as with hot-wires in the case of neutral flow. The free-stream wind speed, measured at a reference point placed 3.1 m upwind of the hill crest at $h = 1$ m, was $U_\infty = 2.5$ ms^{-1}.

The undisturbed velocity profile could be fitted to a logarithmic profile up to at least $h = 1$ m, indicating that $\overline{u_1' u_3'}$ was uniform throughout the measurement area. However, no useful measurements could be taken below $h = 0.05$ m due to the flow features created by the individual 0.02-m high roughness elements. The inner region, 0.059 m deep for the shallow hill and 0.031 m deep for the steep hill, was therefore off-limits.

Numerical simulations by Ross et al.

Ross et al. simulated the flow over the shallow and steep hills with a CFD finite-difference code, using four different turbulence closures:

a. A standard one-equation mixing-length closure

b. A mixing-length closure modified to reduce the over-estimation of turbulent transport in regions of high velocity gradients, such as wakes

c. A full second-order closure with a transport equation for the Reynolds stresses

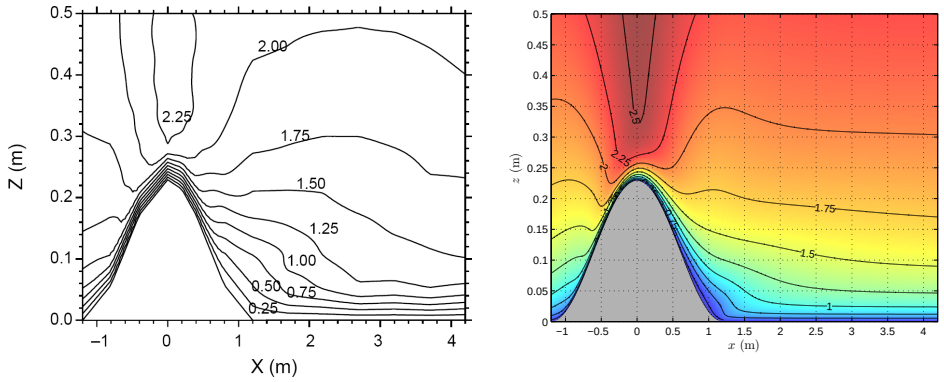
d. A closure with a prescriptive equation for the dissipation, including the mixing length equation of closure [a].

A full description of these closures can be found in the original paper. Closure [b] was most successful at predicting the wind speed perturbation above the hill crest, while [c] and [d] performed somewhat better in the wake region. Numerical stability was a serious issue, however: convergence could not be obtained with closure [c] in the case of the steep hill.

RAMSIM calculations

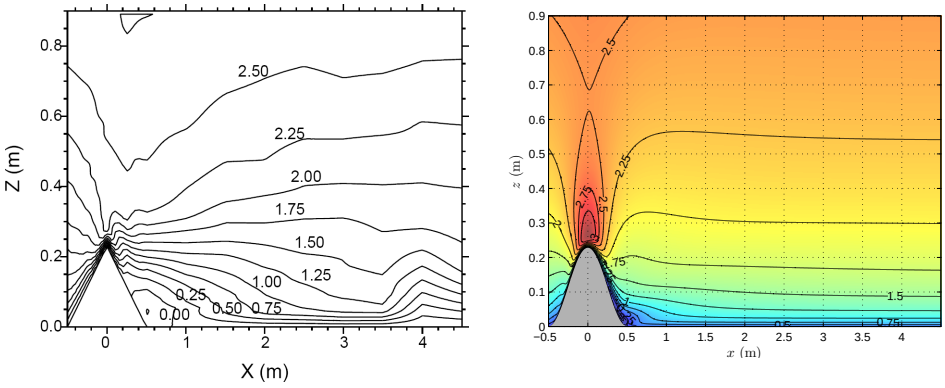
The shallow and steep hills were created at high resolution (1.4 mm), buffered with $L_{\text{buf}} \approx 50H = 11$ m, low-pass filtered with $\sigma = 0.02$ m, and resampled at a lower resolution (1.1 cm) with 2048 points. We had $kz_{0\min} = 6.3 \times 10^{-4}$, well within the range of numerical stability discussed in Section 5.5.

Contour plots of the measured horizontal velocity and of that calculated by RAMSIM are compared on Figure 23 for the shallow hills, and on Figure 24 for the steep hills. Vertical profiles of the relative wind speed perturbation $(u - u^{(0)})/U_\infty$ over the hill crest are shown on Figure 25; measurements are compared to calculations with RAMSIM as well as with the LINCOM model (introduced on page 26). Also shown is one of the numerical simulations by Ross et al.; only the best one is plotted, namely that with closure [b]. Figure 26 compares the surface pressure on the shallow hill calculated by RAMSIM to Ross et al.'s measurements and numerical simulation with closure [b].



(a) wind-tunnel data (b) RAMSIM calculation

Figure 23: Contour plot of u (m/s) over the shallow hill. Height is exaggerated by a factor of 8.6.



(a) wind-tunnel data (b) RAMSIM calculation

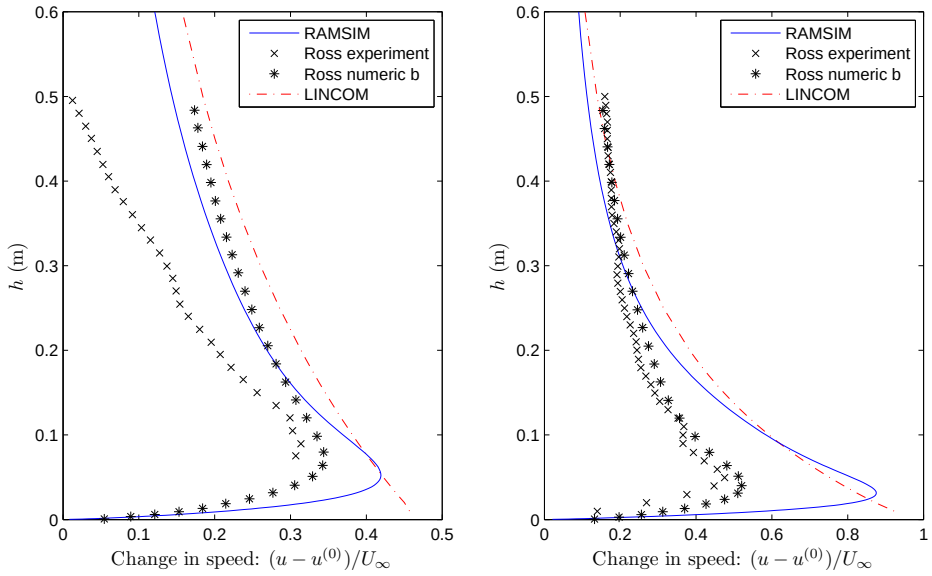
Figure 24: Contour plot of u (m/s) over the steep hill. Height is exaggerated by a factor of 4.4. The colour scale is not the same as on Figure 23.

The maximum speed perturbation over the shallow hill is measured at $h = 0.09$ m above the hill crest; here, RAMSIM overestimates the perturbation by 22%, and predicts an even larger perturbation at $h = 0.05$ m, which is likely an artifact, though no useful measurements are available this close to the ground. LINCOM predicts a larger speed change than both other models throughout almost the entire available domain. Ross et al.'s numerical simulations over-predicts the speed perturbation at $h = 0.09$ m by 9% (closure [b]) to 19%. Simulations with closures [c] and [d] (not shown) predict a larger speed-up at $h = 0.05$ m, much like RAMSIM does.

Higher up, the perturbation profiles yielded by RAMSIM and all of Ross et al.'s numerical simulations converge, but depart markedly from the measured perturbation, which dies out around $h = 0.5$ m. Since this discrepancy does not occur over the steep hill, doubt may be cast over the conditions in which the shallow-hill speed measurements were performed.

There is good agreement between the pressure measurements and RAMSIM's results on the upstream slope (Figure 26). At the hill crest, however, RAMSIM underestimates the pressure reduction by 18%, while Ross et al.'s numerical simulation overestimates it by 20%. Both models, especially RAMSIM, predict a more rapid recovery of the pressure than is measured in the wake region.

With a maximum slope of 0.72, the steep hill is a far cry from the gentle slopes over which linear models are strictly valid, and this does indeed affect RAMSIM's



(a) Shallow hill (b) Steep hill

Figure 25: Hilltop vertical profiles of the speed change relative to the free-stream velocity U_∞.

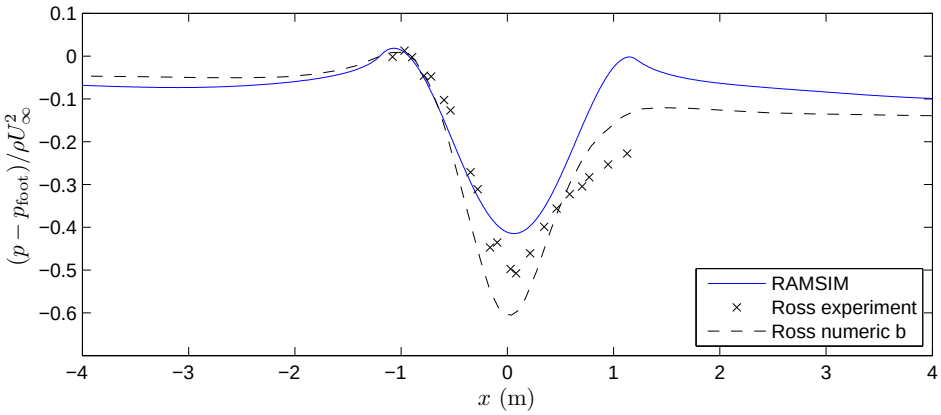


Figure 26: Surface pressure over the shallow hill: RAMSIM results, wind-tunnel data, and numerical simulation [b] by Ross et al. The pressure is normalized by $\frac{1}{2}\rho U_\infty$ and is relative to the pressure p_{foot} at the upstream foot of the hill.

results. The maximum speed perturbation over the steep hill's crest, measured at $h = 0.09$ m, is overestimated by 70% by RAMSIM, which again predicts an even larger speed-up closer to the ground, 84% larger than the maximum measured speed-up. LINCOM's over-prediction at $h = 0.09$ m is of 58%, but its speed-up grows as the ground is approached. Oddly, the numerical simulations by Ross et al. seem to agree better with the profiles over the steep hill than with those over the shallow hill.

Recirculation occurs in the lee of the steep hill. The measured flow reversal is not steady, but on average, the locus of $u = 0$ extends from the hill's lee side at almost half the hill height, downwind to $x = 0.85$ m. RAMSIM calculates a much reduced flow reversal zone, with the locus of $u = 0$ extending only to $x = 0.55$ m, just beyond the foot of the hill, and to about half the height of the measured curve. The perturbation in the wake region as a whole is underestimated by RAMSIM,

which predicts a more rapid recovery of velocity than measured, especially in the case of the steep hill.

Ross et al.'s numerical simulations perform better in the wake region, notably behind the steep hill, where they predict recirculation of almost the right size and magnitude — except when using closure [c], which could not converge.

Ross et al. do not comment on the odd flow feature at $x = 4$ m behind the steep hill, where the flow suddenly slows down.

6.3 Hjardemål escarpment

Hjardemål escarpment in Northern Jutland consists of a 30-degree ramp, cutting almost linearly through agricultural land, rising 16 meters from the post-glacial seabed (now slightly above sea level) up to a wide, nearly-flat plateau. This escarpment is as close an approximation to two-dimensional terrain as one gets in nature, and therefore, when the wind blows perpendicularly to the escarpment, offers an excellent test case for our two-dimensional flow model. A wind measurement campaign was conducted on Hjardemål escarpment in September and October 1989. The complete description of the site, instrumentation, and results is available in the data report produced by Emeis, Courtney, Højstrup and Jensen (1993).

Uphill and downhill flow

From a wind energy point of view, the location of greatest interest for siting wind turbines would be the top of the escarpment, due to the anticipated speed-up when the wind flows uphill. It is therefore important that flow models be able to accurately calculate this speed-up effect. In an orographically more complex terrain, however, such an advantageous location may very well find itself in the lee of some other orographic feature when the wind changes direction. Hence it is also of interest to evaluate the performance of wind flow models in the case of downhill flow at Hjardemål, even though one would never actually put a wind turbine in the lee of an escarpment.

Experimental setup

The escarpment runs approximately in the southeast-to-northwest direction (152°-332°). On the lower side, a variety of hedges, ranging from a one-meter-high stone wall to a row of thin, five-meter-high trees, run almost perpendicular to it (along 236°). An array of eleven meteorological masts was set up, to avoid unexpected effects from the hedges, along a line running in this same direction, halfway between two hedges (40 m away from each), and stretching from 300 m downhill to 400 m uphill of the escarpment. When the wind direction is in the interval 229-247° (flowing uphill) or 47-64° (flowing downhill), none of the masts are placed directly downwind of any part of the hedges. The masts carried cup and sonic anemometers, wind vanes, and temperature sensors, placed at heights ranging from 2 m to 30 m above ground level. The sonic instruments measured the three components of the Reynolds stress τ_{ii} needed to compute the TKE (Equation 9). The experiment took place immediately after harvest; the fairly uniform roughness was $z_0 = 6$ cm, as determined from a logarithmic wind profile measured at a reference mast.

Experimental data

Wind data was recorded for two weeks (weeks 40 and 41), during the periods when the wind direction diverged by less than 20-30° from the perpendicular to the es-

carpment; 26 such continuous periods, termed "runs", were recorded. Among these data collected by Emeis et al., time intervals that had stationary meteorological conditions for at leat 30 minutes, favourable wind directions (perpendicular to the escarpment and unobstructed by hedges), and neutral thermal stability, were subsequently selected by Jørgensen et al. (2006). Six such intervals were identified: four in which the wind flows uphill, and two with downhill flow. These intervals provide the basis for comparison with RAMSIM calculations presented here.

RAMSIM calculations

A digitized height profile of the escarpment, sampled every 0.33 m, was derived from the information in Figure 2.2 and Table 2.1 of the 1993 data report, covering a total distance of 700 m. The terrain was buffered with $L_{\mathrm{buf}} = 1000$ m, low-pass filtered with $\sigma = 2$ m, and re-sampled with 4096 points. Many wavenumbers were used in order to resolve the relatively sharp foot and top of the escarpment accurately, perhaps excessively so: the difference between the measured and the re-sampled terrain height was at most 30 cm, thus having the same negligible effect on the flow as the presence or absence of a large stone would have. We had $kz_{0\min} = 7 \times 10^{-5}$, well within the range of numerical stability discussed in Section 5.5.

RAMSIM results for both the uphill and downhill flows will now be compared to the field measurements using cup anemometers, as well as to Jørgensen et al.'s (2006) results using the WAsP flow model as well as EllipSys, Risø's in-house non-linear CFD model introduced on page 28. To facilitate comparison, all wind speeds are normalized by u_{ref}, the velocity at the location of the most upwind mast at a height above ground level of $h = 10$ m. Correspondingly, all TKEs are normalized by u_{ref}^2.

Uphill flow The uphill flow remains attached, except perhaps right at the foot of the escarpment, where RAMSIM predicts a small recirculation bubble about 1 m in size. The measured and calculated vertical velocity profiles of the uphill flow are shown on Figure 27. The wind speeds calculated by the three models are all in general agreement with the field measurements. The largest discrepancy occurs right at the top of the escarpment ($x = 0$). Here, the wind speed measurements at $h = 10$ m span a range of 5%, and EllipSys falls within this range. WAsP exceeds the measurements by 6 to 11%. RAMSIM exceeds them by a lesser margin: 3 to 7 %. RAMSIM predicts an excessively high speed-up at $h = 2$ m, where it overestimates the wind speed by about 20%. Such low heights are far below the hub height of almost any wind turbine, and thus of limited significance. However, if we scaled up the hill by an order of magnitude, the overestimate would be, to a first approximation, located at $h = 20$ m; this is at the marginally low end of hub heights for modern turbines, and is of the same order of height as most meteorological masts used for wind resource assessment measurements (10 to 40 m).

The wind speed profile at 10 m a.g.l. calculated by EllipSys largely falls within the data points. Both RAMSIM and WAsP overestimate by about 5% the wind speed above the foot and the top of the escarpment (Figure 28).

In order to compare TKE calculations with the measurements from the sonics, an averaging time must be chosen for the latter. This choice is somewhat arbitrary: a shorter time will leave out some of what is considered turbulence, and a longer time will include the effects of some of what is considered the mean flow. We had at our disposition the TKE measurements with averaging times ranging from 1 to 10 minutes. Even between these two extremes, however, there was little change in the shape of the TKE profiles at 10 m a.g.l. The 10-minute values were generally

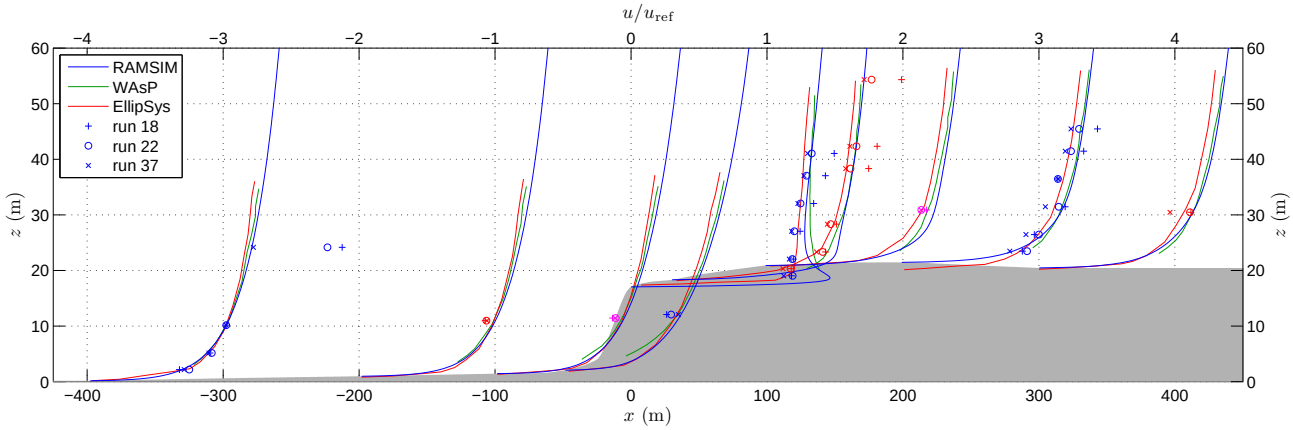


Figure 27: Uphill flow: vertical profiles of the normalized horizontal wind velocity. RAMSIM is compared with cup anemometer measurements, as well as WA^{S}P and EllipSys results. The profiles should be read as follows: At a given height above the terrain, the normalized wind speed is the difference between the values of the upper axis at the given height and at the bottom of the profile, where the speed is zero. The grey area represents the terrain. Height is exaggerated by a factor of 5 relative to horizontal distance.

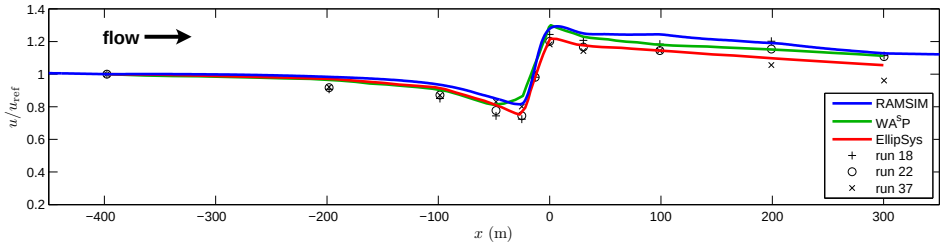


Figure 28: Uphill flow: profiles of the normalized horizontal wind velocity at 10 m a.g.l.

15% larger than the 1-minute ones, except run 22, for which this proportion was 36%. We chose an averaging time of 5 minutes. Keeping the above differences in mind, comparisons are possible.

The level of TKE given by the analytical zero-order solution at the unperturbed upstream reference point is about 20% greater than EllipSys' value, which falls close to the data points. However, the perturbation around the escarpment itself seems to be better captured by RAMSIM than by EllipSys, which calculates an increased TKE beyond the top of the escarpment, while measurements show a rapid recovery (Figure 29).

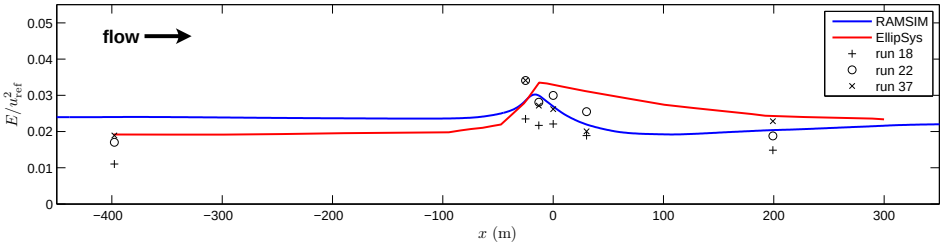


Figure 29: Uphill flow: profiles of the normalized TKE at 10 m a.g.l.

Downhill flow The downhill flow is more difficult to model, as it detaches substantially in the lee of the escarpment. RAMSIM predicts a recirculation region up to 4 m deep and 20 m wide, creeping up about 60% of the slope (Figure 30). Though no direct observations of this region are available, other measurements

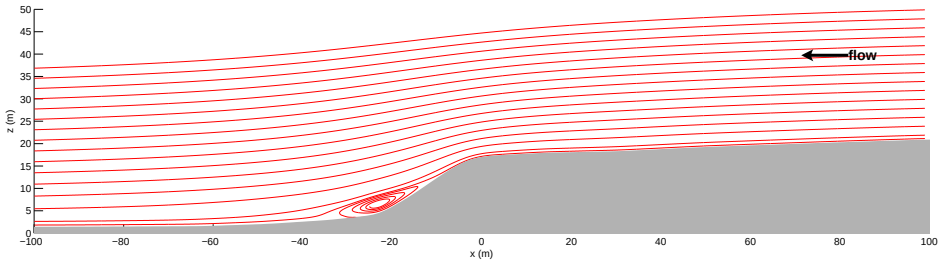


Figure 30: Streamline plot of the downhill flow calculated by RAMSIM, highlighting the recirculation region in the lee of the escarpment.

as well as the EllipSys simulation indicate that the actual recirculation region is larger.

The measured and calculated vertical velocity profiles of the downhill flow are shown on Figure 31. RAMSIM largely agrees with the field measurements at the

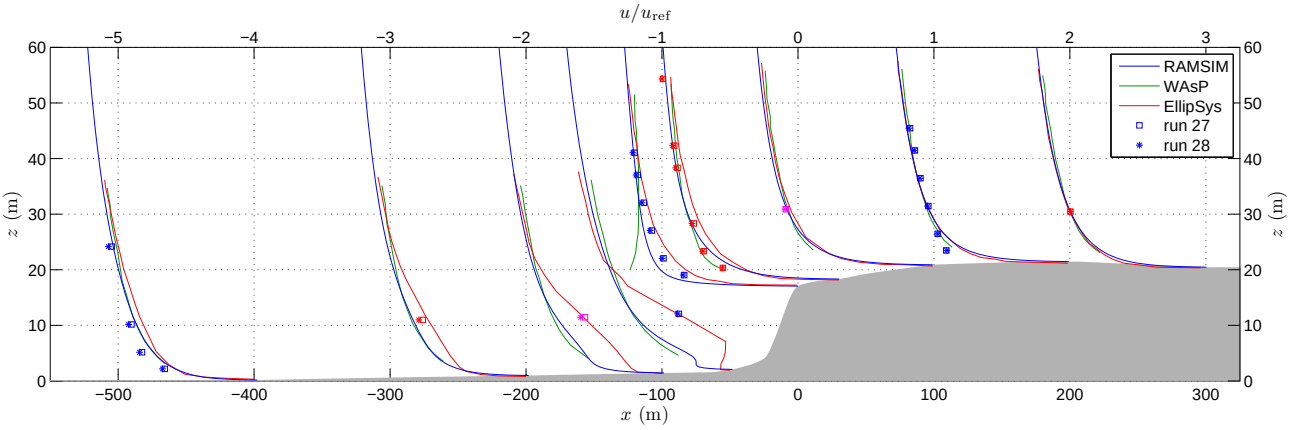


Figure 31: Same as Figure 27 but for the downhill flow (flow direction is from right to left).

top of the escarpment (where one would place a wind turbine), but not in the immediate lee of the escarpment, where the speed reduction is significantly underestimated; this is also visible on profiles taken at a fixed height of 10 m (Figure 32). The profiles given by EllipSys, on the other hand, agree completely with the measurements in the lee-side measurements, and hint at a larger recirculation region than that predicted by RAMSIM. The sharp bend at $h = 5$ m in the profile at $x = -50$ m indicates that this point is close to the trailing end of a recirculation bubble, whereas RAMSIM's bubble does not extend beyond $x = -35$ m.

RAMSIM still performs somewhat better than WAsP, especially right at the top of the escarpment, where it agrees with the field measurements, while WAsP predicts a large, non-existent speed-up effect below $h = 10$ m. RAMSIM also predicts a slightly more pronounced speed reduction near the foot of the escarpment than WAsP does, and predicts the presence of a recirculation zone, while WAsP assumes attached flow throughout the domain.

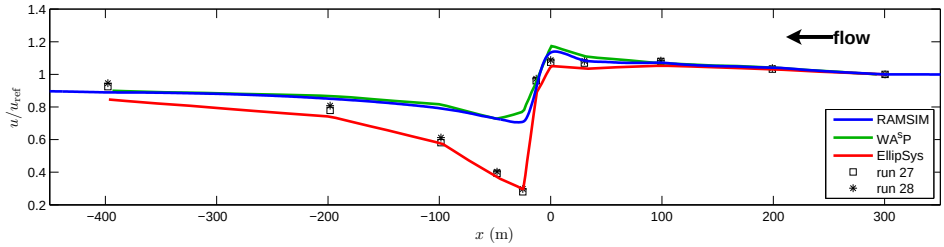


Figure 32: Downhill flow: profiles of the normalized horizontal wind velocity at 10 m a.g.l.

The level of TKE at the upstream reference point is calculated correctly by both RAMSIM and EllipSys (Figure 29). Just upwind of the escarpment, between

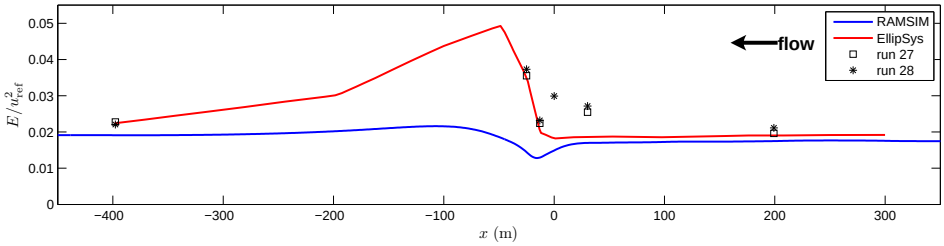


Figure 33: Downhill flow: profiles of the normalized TKE at 10 m a.g.l.

$x = 30$ m and 0 m, an increase in TKE is measured, but neither model calculates this; in fact, both have a slight downward trend in this region. Jørgensen et al. (2006) attribute the measured increase of TKE to the concave curvature of the terrain upstream of the top of the escarpment; such an effect has been reported by, for instance, Jensen (1983). Jørgensen et al. explain the non-responsiveness of EllipSys to this effect by the law-of-the-wall boundary condition employed, which requires the horizontal pressure gradient to be zero, so that the concavity-induced pressure gradient near the surface is not accounted for in the momentum balance. RAMSIM does allow a non-zero horizontal pressure gradient at the surface, but offers no improvement over EllipSys. It appears doubtful, however, that such a mild concavity as occurs near the top of the escarpment (see the ground profile on the exaggerated height scale of Figure 31) would produce a TKE perturbation one-half as large as that produced by the much larger concavity and the separated flow at the foot of the escarpment. The Askervein terrain (discussed in Section 6.4) also presents small concavities, but no similar effect on the TKE is observed. This raises the possibility of measurement artifacts in the Hjardemål case.

At $x = -13$ m, the measured TKE returns to its baseline value. Here, EllipSys matches the measurements, but RAMSIM registers a 37% drop below this level. Around $x = -15$ m, the 10 m a.g.l. line begins to fall in the shadow of the escarpment, where the flow separates and much TKE is created; hence we see a sharp rise in the measured TKE, matched by that calculated by EllipSys. Here, RAMSIM registers an small increase in TKE, but starting from a reduced value; hence there is a qualitative failure to predict the heightened level of TKE.

These errors are likely due to the rising importance of non-linear effects as the escarpment top is passed and flow separation sets in; the flow separation calculated by RAMSIM occurs much below the $h = 10$ m line. Moreover, RAMSIM is inherently constrained to zero-mean perturbations along a line of fixed $\acute{z}$ (to first order, $\acute{z} \approx h$), and so it is forced to reduce the TKE somewhere (e.g. around the

escarpment top) if it is to increase it elsewhere (e.g. in the wake region). The measured TKE perturbation is, however, exclusively positive.

6.4 Askervein hill

The Askervein study, known in certain circles as "the mother of all wind experiments", is the most extensive wind flow measurement campaign undertaken to this date. Carried out as a two-phase, multi-national research project in 1982 and 1983, it involved the deployment of over fifty meteorological masts on Askervein hill, located on the island of South Uist in the Outer Hebrides of Scotland. An overview of the project was published by Taylor and Teunissen (1987), including an extensive description of the setup and instrumentation, and a summary of the data collected.

Askervein hill clearly represents three-dimensional terrain; hence three-dimensional effects do affect wind flow. However, since the hill is very elongated, one can expect two-dimensional effects to dominate when the wind blows perpendicularly to the hill's major axis, thereby offering the possibility of comparing our two-dimensional calculations to field measurements.

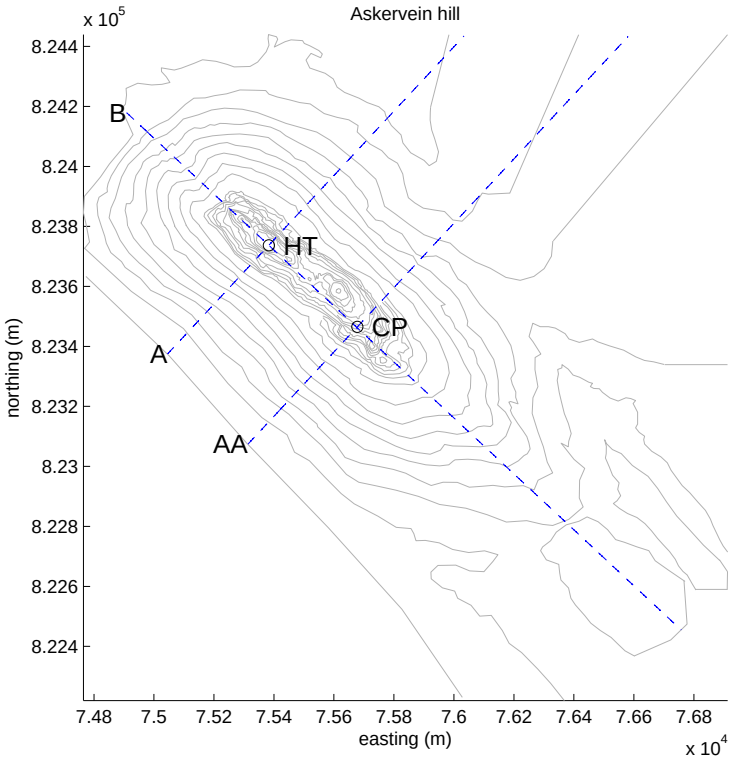


Figure 34: Topographical map of Askervein hill. The height contours spacing is 10 m near the base of the hill, and is refined down to 1 m at the top of the hill. Lines A, AA, and B, as well as points HT and CP, are indicated.

The hill

Askervein hill is 116 meters high, and presents itself as an elongated, smoothed-out ridge, with mostly cigar-shaped height contours (Figure 34), and with a full width at half-maximum of approximately 500 m along its short axis and 1300 m along its long axis. Height profiles taken perpendicularly to the long axis are eerily Gaussian

(Figure 35), with maximum slopes of about 0.30 to 0.38. The terrain southwest of the hill is largely flat for several hundred meters, thus offering space for an undisturbed flow to develop.

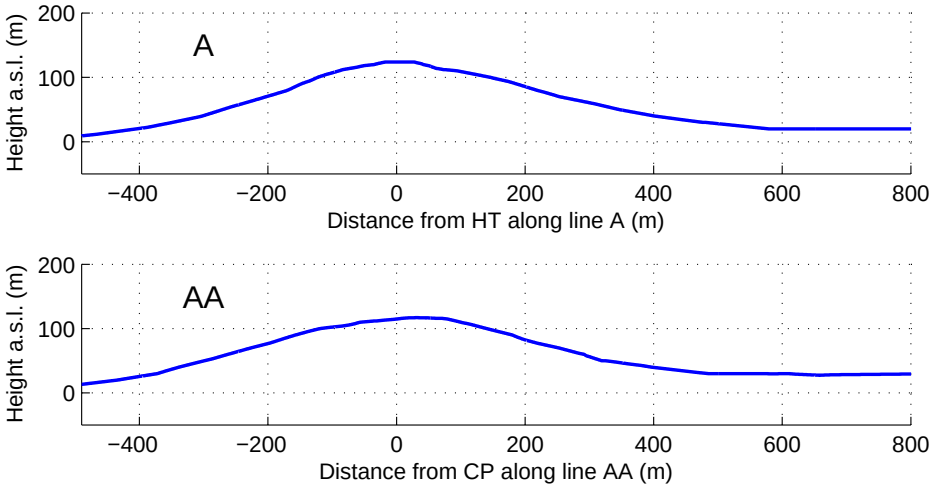


Figure 35: Height profiles of lines A and AA.

Experimental setup

Most of the masts deployed were 10 m high and carried a single cup anemometer, but a few were higher (up to 50 m) and carried additional cup anemometers at up to seven different heights, and several were equipped with instruments for three-component turbulence measurements. In the 1982 phase, 10-m masts were placed along lines A and B, and also line AA in the 1983 phase (Figure 34), allowing for the measurement of the spatial variations of mean wind speed at fixed height. Vertical profiles were measured at points HT (hilltop) and CP (centrepoint).

RAMSIM calculations

A digitized height contour map of Askervein hill, formatted for use with WA^{S}P was retrieved from archives at the Risø Wind Energy Department. Terrain height profiles about 1.5 km in length were interpolated from this map along lines A and AA, buffered with $L_{\mathrm{buf}} = 1000$ m, low-pass filtered with $\sigma = 10$ m, and resampled with 2048 points. The roughness length was taken to be $z_0 = 3$ cm (consistent with most of the literature, e.g. Salmon et al. 1988), yielding $kz_{0_{\mathrm{min}}} = 2.5 \times 10^{-5}$, still within the range of numerical stability.

RAMSIM calculations are compared to measurements taken during from run TU-03B of the 1983 measurement campaign, during which the wind direction was 210°, i.e. arriving from the flat area southwest of the hill, and flowing more or less perpendicularly (77°) to the hill's major axis. We will also compare results to those of Beljaars et al.'s (1987) linear MSFD model with $E - \varepsilon$ closure, and of Castro et al.'s (2003) non-linear finite-volume simulation with RNG $E - \varepsilon$ closure with their highest-resolution grid ($155 \times 155 \times 31$).

The fractional speed-up ΔS is defined as the fractional change in horizontal

wind speed compared to the undisturbed wind profile:

$$\Delta S(x,h) = \frac{u_1(x,h) - \acute{U}_1^{(0)}(h)}{\acute{U}_1^{(0)}(h)}$$

where h is the height above ground level. RAMSIM's calculated vertical profile of the fractional speed-up at point HT agree better with the measurements than Beljaars et al.'s and Castro et al.'s calculations (Figure 36). At point CP, RAMSIM

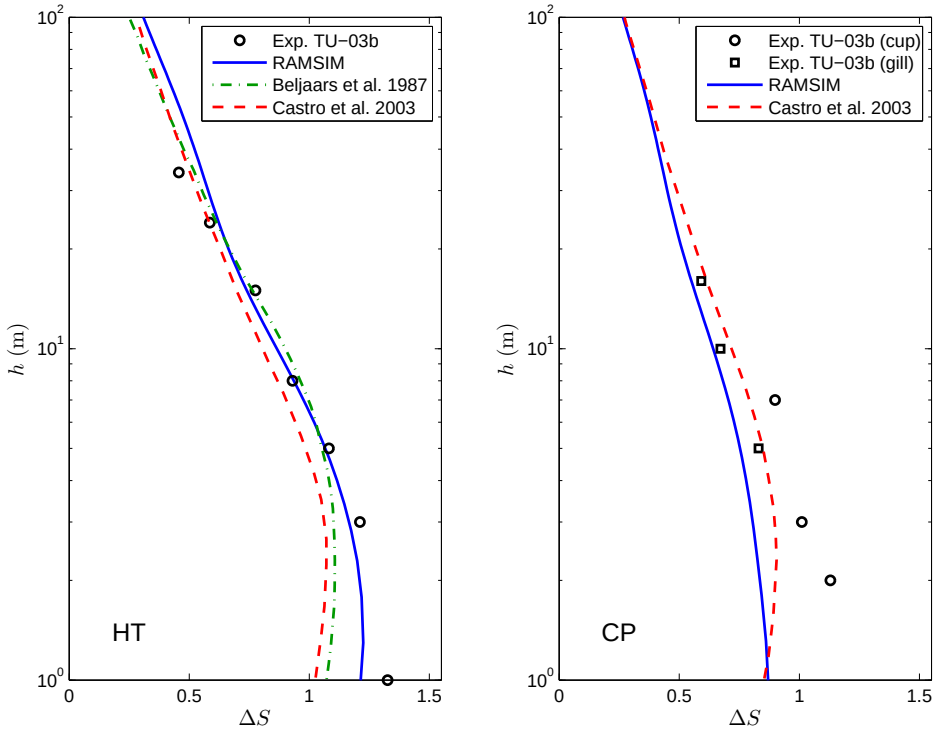


Figure 36: Vertical profiles of the fractional speed-up: Calculations by RAMSIM are compared to those by Beljaars et al.'s (1987) linear MSFD model with $E-\varepsilon$ closure and by Castro et al.'s (2003) non-linear finite-volume simulation with RNG $E-\varepsilon$ closure, as well as to run TU-03B measurements over points HT (cup anemometers) and CP (cup and Gill anemometers).

under-estimates the speed-up at CP below $h = 7$ m; so does Castro et al.'s non-linear model, but by a lesser amount.

The actual roughness length near the hilltop has been reported to be lower than the value of 3 cm prevailing in the surroundings, possibly as being little as 1 cm (Zeman and Jensen 1987). This would account for the speed-up underestimates by the three models. Indeed, Castro et al. conducted further simulations with a correction for the variable roughness, and obtained better agreement with the measurements below $h = 10$ m without affecting the rest of their profile.

The speed-up profiles calculated by RAMSIM and at 10 m a.g.l. along line A are similar to those of the linear MSFD model: both are close to the measurements on the upwind side and crest of the hill, but not in the lee of the hill, where the speed reduction is underestimated in the region starting 100 m beyond the hill crest and extending at least down to the foot of the hill (Figure 37). The wind speed in this region is overestimated by a factor of up to 2.2 at $x = 391$ m. In contrast, Castro et al.'s non-linear calculations capture the speed reduction in the wake region, perhaps excessively so: the wind speed is *under*estimated by about the same factor. Along line AA, results are less differentiated. RAMSIM almost

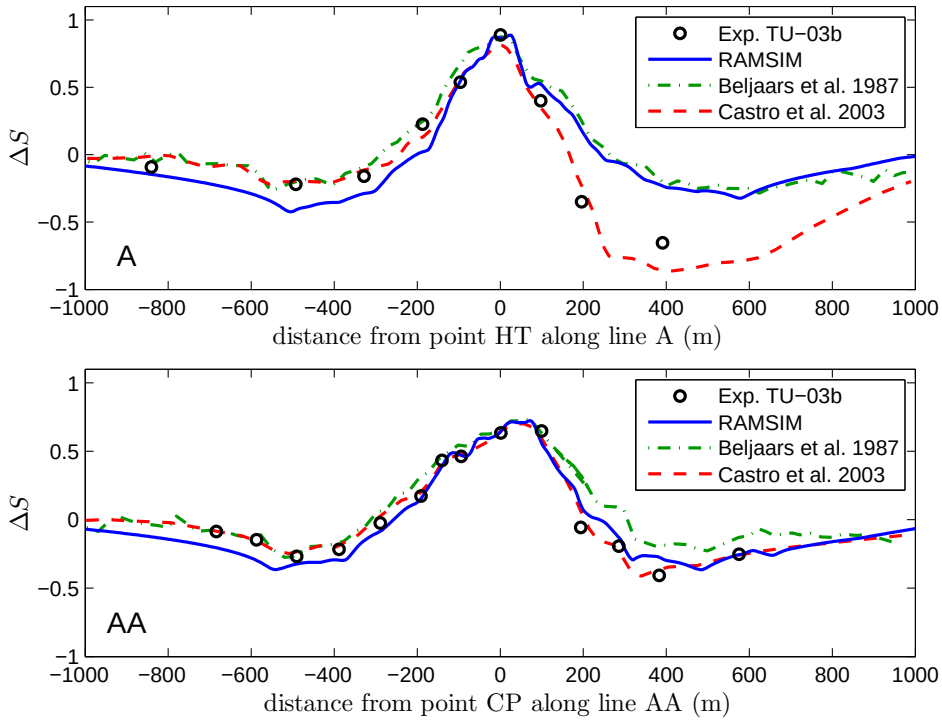


Figure 37: Profiles of the fractional speed-up at 10 m a.g.l. along lines A and AA: Calculations by RAMSIM, Beljaars et al.'s (1987) linear MSFD model with $E-\varepsilon$ closure, and Castro et al.'s (2003) non-linear finite-volume model with RNG $E-\varepsilon$ closure, compared to run TU-03B measurements (cup anemometers).

matches the non-linear model's accuracy on the lee side, while the linear MSFD model overestimates the wind speed by about 35%.

Irregular, small-scale fluctuations are observed on the calculated wind speed profiles at 10 m a.g.l.; these will be discussed in Section 7.3.

Modelling the TKE is more difficult. Castro et al.'s non-linear calculations agree qualitatively, but not quantitatively, with the observations along line A; too few measurements are available on line AA for a meaningful comparison (Figure 38). More recent non-linear simulations by Eidsvik (2005) and Undheim et al. (2006) (not shown) have a generally decreased level of TKE compared to Castro et al.'s, thus agreeing better with the measurements upstream of the hill crest ($x < 0$ m), but less well with those on the lee side ($x > 0$ m), though this does not seem to affect their accurate calculation of the fractional speed-up in both regions.

As in the case of Hjardemål escarpment, RAMSIM fails to qualitatively model the TKE perturbation when the wind flows past the hill crest and over a downward slope. This is likely due to the importance of non-linear effects in this region, and to the fact that RAMSIM is constrained to a zero-mean TKE perturbation.

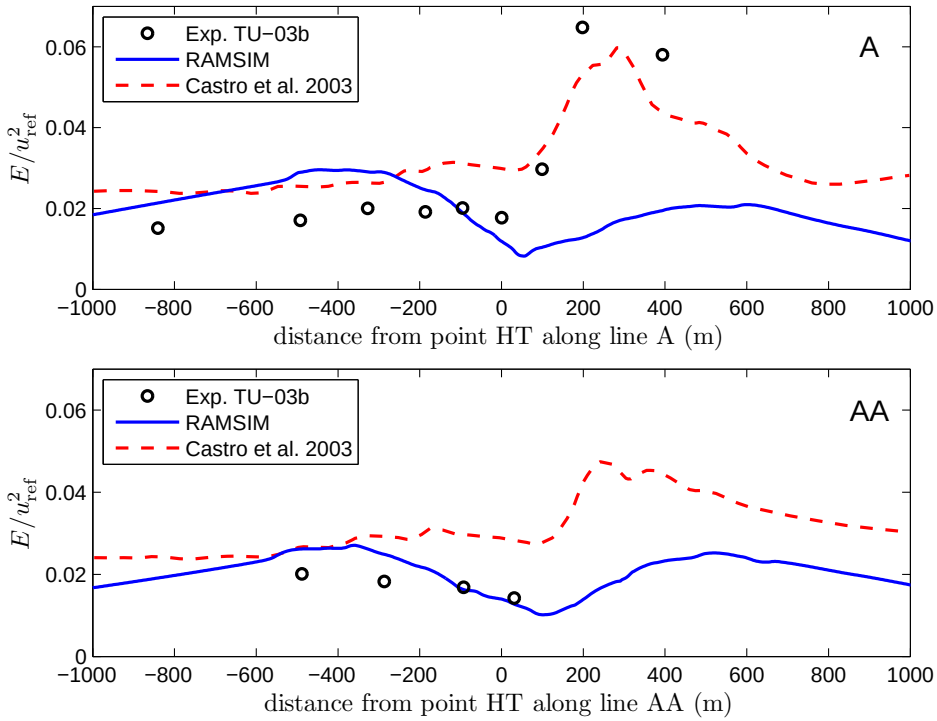


Figure 38: Profiles of the TKE at 10 m a.g.l. along lines A and AA: Calculations by RAMSIM and Castro et al.'s (2003) non-linear finite-volume model with RNG $E-\varepsilon$ closure, compared to run TU-03B measurements.

7 Discussion

7.1 Performance assessment

Computational load

The generation of the RAMSIM look-up table on a 2-GHz PC takes 2 minutes, which in and of itself is rather quick. Once generated, the LUT can be used to calculate the flow over any terrain extremely quickly. Calculating the flow field on a 2048×150 grid, which provides more than sufficient resolution for any wind resource assessment, takes under 3 seconds of CPU time. This spatial resolution (3 m in the case of Askervein) is perhaps excessively fine: non-linear CFD models usually employ far coarser grids, by a factor of 10 or so in the horizontal direction; RAMSIM calculations on comparable grids would correspondingly be much faster.

The planned extension of our model to three dimensions will of course cause the computational time to increase, for three reasons. Firstly, there will be two more variables to interpolate from the 3D LUT, namely Ω and Ω'. These will then be used in additional calculations to determine the two horizontal velocity components.

Secondly, matching a measured wind direction θ_m at a given point in the domain may require $n_\mathrm{it} = 3$ or 4 iterations with different forcing stress angles θ_f, thus multiplying the computation time by n_it, as explained in Section 5.2 (page 62). However, calculating the flow field for, say, $n = 12$ different θ_m spread across the 360° wind rose, as is usually required in wind resource assessment, will not increase the computation time by a factor $n_\mathrm{it}\, n \approx 48$, but only by a factor as little as 3, up to at most 10 or so. This is in part due to the effect of mirroring, explained in Section 5.7 (page 72).

Thirdly, and most importantly, the calculation domain will have an additional horizontal dimension. This will roughly multiply the calculation time by N_y, the number of wavenumbers in this dimension (the FFT computation time, proportional to $N_y \log N_y$, makes up only a small fraction of the total computation time). Calculations for a single forcing stress angle θ_f on a $2048 \times 2048 \times 150$ grid would thus require an estimated 2 hours. A coarser $512 \times 512 \times 50$ grid, still finer than the finest grids used by Castro et al. (2003) over Askervein hill, would result in computational times of the order of 3 minutes for a single value of θ_f.

RAMSIM is constrained to calculating the entire flow field at once, and as such is not as quick for single-point calculations as WAsP, whose architecture is designed precisely for this kind of task (WAsP takes only a small fraction of a second). But the calculation of the entire flow field is precisely what is required when producing resource maps, i.e. a grid of the wind resource at a given height over a large area. This can take WAsP a few hours (for a 3D terrain) if a high resolution is required, since it repeats its calculation procedure in a point-by-point fashion. RAMSIM promises to be much better suited to this task.

Upwind slope and hilltop

There is generally good agreement between RAMSIM results and measurements over upwind slopes and at the top of hills or escarpments, where there is most in placing wind turbines. This is true despite the fact that the assumption of linearity appears to be strained, with measured fractional speed-ups in excess of 100%. This observation is consistent with previous experience with linear-model simulation of the flow over Askervein hill, reviewed by Walmsley and Taylor (1996), as well as over two other hills with moderate slope that were the subject of wind

measurement campaigns: Kettles hill (Salmon, Teunissen, Mickle and Taylor 1988) and Blashaval (Mason and King 1985).

RAMSIM has a tendency, however, to predict a spuriously large speed-up near the ground at the hill top; though not as large as that predicted by WAsP over Hjardemål escarpment. This should not represent a major problem in practice: in both full-scale test cases examined, the spurious speed-up region was far below the hub heights of modern wind turbines; calculated speed-up maxima occurred around $h = 2$ m at the top of Hjardemål escarpment and $h = 3$ m over the top of Askervein hill. Either hill would have to be scaled up by an order of magnitude for these spurious maxima to reach the hub height of the smallest turbines commonly erected today.

An important exception is if the measurement used to normalize the calculated flow field is taken precisely in a spurious speed-up region; this would result in an excessive down-normalization of the predicted wind resource at hub height. Most meteorological masts used for wind resource assessment measurements are 10, 20, or occasionally 40 metres high. Hjardemål or Askervein would have to be scaled up by a factor of only two or three for the smallest (and, unfortunately, most common) of these masts to be affected by the top of the spurious speed-up region.

Lee side and wake

Since wind turbines are unlikely to be placed in regions of reduced wind speed, model accuracy in wake regions is not as critical as in regions of anticipated speed-up. As with the previous linear models reviewed in Section 1.6, the wind speed in the wake region is much overestimated by RAMSIM, and recovery occurs more rapidly than is measured. However, contrary to other linear models, to the best knowledge of the author, RAMSIM is capable of predicting the occurrence of recirculation. The dimensions of the recirculation region are underestimated, usually by about half.

RAMSIM does not capture the lee-side increase in TKE measured by sonic instruments and calculated by non-linear flow models. These errors can be attributed in part to the importance of non-linear effects in the wake region. Perhaps more importantly, as a linear model, RAMSIM is inherently constrained to zero-mean perturbations at a given $\acute{z}$. This may be a good approximation for the velocity, but it is not realistic in the case of the TKE, since the actual orography-induced TKE perturbation is almost always positive, i.e. an increased TKE compared to the flat-terrain baseline. A non-linear extension to our model (discussed below) may be necessary to allow for a DC component to appear in the perturbation.

Recirculation implies that the flow perturbation is locally greater than the mean flow, and so the linear approximation becomes suspect — so does it when the speed-up becomes very large at hilltops. We are still confident that the behaviour of the flow remains somewhat physical, even in recirculation regions, since the first-order perturbation does contain some physics, most notably continuity and incompressibility, which forces the velocity field to respond realistically to the presence of orography.

Referring back to Section 2.5, the linear approximation is valid if higher-order perturbations have less influence on the flow than the first-order perturbation:

$$f_2 + f_3 + \ldots \quad \ll \quad f_0 + f_1 \tag{118}$$

Since by definition we do not calculate the higher-order terms, we have no direct way of determining whether this condition is satisfied. The relative size of the zero- and first-order contributions is thus usually taken as clue on the behaviour of higher-order contributions: $f_1 < f_0$ suggests that the series is close to convergence and that the perturbation decreases in importance as one climbs in order ($f_2 < f_1$,

etc.), and conversely, $f_1 > f_0$ suggests that $f_2, f_3, \ldots$ might be even greater. These, however, are mere indications: $f_1 < f_0$ does not guarantee linearity, nor does $f_1 > f_0$ necessarily invalidate linearity. The validity of the linear approximation can only be tested by comparing calculations with those of non-linear models and determining the influence of non-linear terms. One way of doing so, and to improve results in regions where we are suspicious of the linear approximation, is by developing a non-linear extension to our model.

Non-linear extension

A possible avenue for the mitigation of the inaccuracies in the wake region and of the overestimated speed-up at hilltops, would be to derive some of the most important non-linear second-order terms, and include them as extra source terms in the first-order equations, in an approach similar to that of Xu and Taylor (1992) and Xu et al. (1994). The equations would be solved iteratively in Fourier space. The starting point would be the linear solution we have now; subsequent iterations would include the non-linear terms, calculated in physical space based on the flow field generated by the previous iteration.

This will cause the user computational time to increase significantly: a new LUT will have to be built for each iteration, though only over the range of kz_0 relevant for the current terrain. One can thus anticipate computation times of the order of 1 minute per iteration for 2D terrain.

A possible scenario would be to have the user interface allow the flow model to operate in two modes: a quick preview mode, in which the the linear model provides a good estimate of the flow field in order tentatively evaluate various arrangements of wind turbines; and a final calculation mode, using the non-linear model for a more accurate prediction of wake effects from hills.

Pressure drag

A further inconvenience in the linear approximation is its inability to account for the force exerted on the flow by the ground via the pressure drag (or form drag). The pressure drag F_p, previously defined in Equation (117) (page 81), is formally a second-order quantity: in the zero-order, unperturbed, horizontally homogeneous case, both the pressure p_0 and the terrain slope $(\partial h_t/\partial x)_0$ are zero. The leading pressure term is p_1, while the terrain slope is purely first-order; and so the leading term of the product of the two is second-order in nature. We can certainly calculate it after the fact, and use it as a diagnostic tool as we did in the sinusoidal hills' test case (Section 6.1). However it does not enter the first-order equations as such, and does not affect our calculated flow.

Following the convention of Wood and Mason (1993), the total force F_{tot} exerted horizontally on the terrain by the flow has contributions from the horizontal components of the pressure force F_p, acting perpendicularly to the ground surface, and of the surface friction F_s, proportional to U_{*0}^2, acting parallel to the surface:

$$F_{\mathrm{tot}} = F_p + F_s$$

In the case of flat terrain, there is no pressure drag, $F_{\mathrm{tot}} = F_s$, and we have the logarithmic profile of Equation (72). The addition of the orography-induced F_p, however, leads to a modified logarithmic profile in the far field:

$$\bar{U}(z) = \frac{U_*^{\mathrm{eff}}}{\kappa} \ln\left(\frac{z}{z_0^{\mathrm{eff}}} + 1\right) \tag{119}$$

where U_*^{eff} is an effective friction velocity (F_{tot} is proportional to the square of U_*^{eff}) and z_0^{eff}, an effective roughness length. As discussed by Wood and Mason

(1993), this relation represents well the horizontally averaged wind speed $\bar{U}(z)$ when z exceeds a few hill heights H above the terrain. This was observed by, for instance, Athanassiadou and Castro (2001) in the sinusoidal hills' test case we examined in Section 6.1.

Seen from $z \gg H$, the orography thus acts as extra surface roughness. As we move up in z, the orography-induced perturbations do not simply die out, but rather settle down at $k = 0$ as a horizontally uniform modification of the undisturbed wind profile. This cannot be captured by our linear model, since all wavenumbers are separated: a terrain component of wavenumber k_a can only produce a (zero-mean) perturbation of wavenumber k_a, and no contribution to the $k = 0$ horizontal average. A non-linear extension will be required to include the effect of the pressure drag, as well as other non-zero-mean effects of the orography, such as the increased TKE.

7.2 Addressing the $kz_0 \to 0$ problem

As our calculation domain becomes large and the roughness length decreases ($kz_0 \to 0$), RAMSIM becomes plagued by numerical instability, as described in Section 5.5. Currently we cannot cover the entire range of non-dimensional roughnesses kz_0 defined in Table 2 (page 62); over two decades are missing.

None of the test cases examined in Section 6 were affected by this particular problem. The closest call was our 1.5 km-long Askervein hill height profiles; after buffering and mirroring, this became a 7 km-long domain with $kz_{0\min} = 2.5 \times 10^{-5}$, which is about one decade above the point where instability sets in (recall Figure 10). However, maps used for WAsP calculations can reach 20 km in length, i.e. 50 km or so after buffering and mirroring. This is already reaching our limit; a roughness length lower than Askervein's 3 cm would put us beyond it.

In order to attack this problem, three prospective approaches present themselves: the first is logistical, the other two are attempts to "quilt" the solution using special limit solutions for the problematic regions in $(k\acute{z}, kz_0)$ parameter space. These approaches have not yet been implemented, and represent the next steps in the development of our flow model.

Increased machine precision

Perhaps the least elegant approach would be to perform our calculations with a greater computer precision. In this manner, we would retain enough significant digits, so that the solutions that are exponentially decaying in $k\acute{z}$ could be better resolved despite their being superposed with the much larger, undesirable, exponentially growing solutions. Our Fortran compiler (Compaq Computer Corporation 2000) can handle complex data types consisting of up to 16 bytes on a PC platform; the 32-byte data type is only available on more specialized hardware, such as the VAX and Alpha family of computers, or systems with a Intel Itanium CPU. However, it has come to our attention that the more recent Intel Fortran Compiler (Intel Corporation 2005) can emulate the extended-precision format in software. In future work, we will attempt a migration to this compiler, in the hope that the 32-byte data type will give us enough precision to cover, in part or in whole, the two missing decades of kz_0.

Reducing the $k\acute{z}$ range of numerical integration using solutions for the limits $k\acute{z} \to 0$ and $k\acute{z} \to \infty$

As argued in Section 5.5 (page 69), the size of our range of numerical integration (from $k\acute{z} = 0$ to $k\acute{z} = k\acute{z}_{\text{top}} = 10$) has a decisive impact on how much numerical

noise is present in the solution: the longer the range, the more the undesirable, exponentially growing solutions end up swamping the interesting, exponentially decaying solutions at the upper boundary. In response to this, one approach would be to reduce the span of $k\acute{z}$ across which the numerical integration must travel, by using special solutions for the limits $k\acute{z} \to 0$ and $k\acute{z} \to \infty$. From $k\acute{z} = 0$ up to some level $k\acute{z} = k\acute{z}_{\text{low}}$, we would use the $k\acute{z} \to 0$ solution, and from $k\acute{z} = k\acute{z}_{\text{top}}$ down to some level $k\acute{z} = k\acute{z}_{\text{high}}$, we would use the $k\acute{z} \to \infty$ solution. Numerical integration of the full equations would be limited to the reduced interval between $k\acute{z}_{\text{low}}$ and $k\acute{z}_{\text{high}}$, where the boundary values would be provided by the limit solutions. This is illustrated schematically in Figure 39. A judicious choice of $k\acute{z}_{\text{low}}$ and $k\acute{z}_{\text{high}}$ would be required to ensure that the limit solutions are still good approximations to the solution of the full equations.

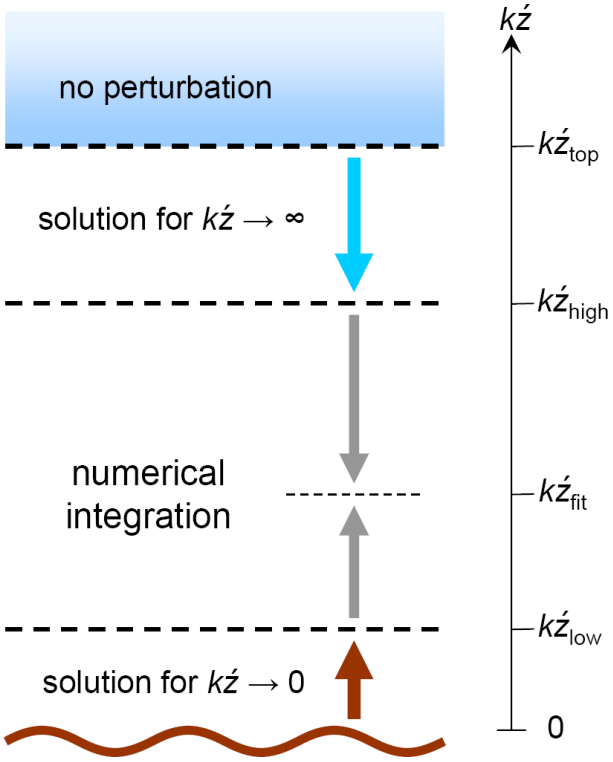


Figure 39: Using solutions for the limits $k\acute{z} \to 0$ and $k\acute{z} \to \infty$.

Limit $k\acute{z} \to \infty$ We have two analytical solutions that obey our upper boundary conditions. The first one is the analytical solution presented in Section 5.6. The second one is the solution to the far-field equation, derived in Section 5.5. A far-field solution $\mathbf{X}_\infty(\acute{z})$ can be obtained from $\mathbf{X}_\infty = \mathbf{T}^{-1}\mathbf{R}$, where $\mathbf{T}$ is the rotation matrix defined in Equation (114), and the analytical solution $\mathbf{R}$ is given in Equation (116).

Limit $k\acute{z} \to 0$ The expression for the zero-order dissipation (Equation 75),

$$\varepsilon_0 = \frac{U_{*0}^3}{\kappa(\acute{z} + z_0)} \tag{120}$$

clearly has the potential to be the seed of numerical difficulties when both $k\acute{z} \to 0$ and $kz_0 \to 0$. In response to this (and also for the previously stated goal of reducing the range of numerical integration of the full equations), one may derive

the equations for the limit $k\acute{z} \to 0$, and attempt to find an analytical solution, or perhaps even integrate them numerically from the surface ($k\acute{z} = 0$) to a low height $k\acute{z}_{\text{low}}$, just high enough to clear the danger zone presented by Equation (120).

Deriving and solving the equations for the limit $kz_0 \to 0$

The fact that kz_0 finds itself isolated in the denominator in a number of terms in our equations may well be the source of our numerical problems when $kz_0 \to 0$. In response to this, one may derive the equations for the limit $kz_0 \to 0$, and investigate both analytical and numerical solutions. Hopefully this limit solution would remain valid long enough to be matched with our current numerical solution at a certain $(kz_0)_{\text{low}}$ where the latter is still numerically stable.

These approaches based on limit solutions are not mutually exclusive; on the contrary, one can envision using them in conjunction, in various regions of parameter space. The resulting solution quilt is illustrated in Figure 40.

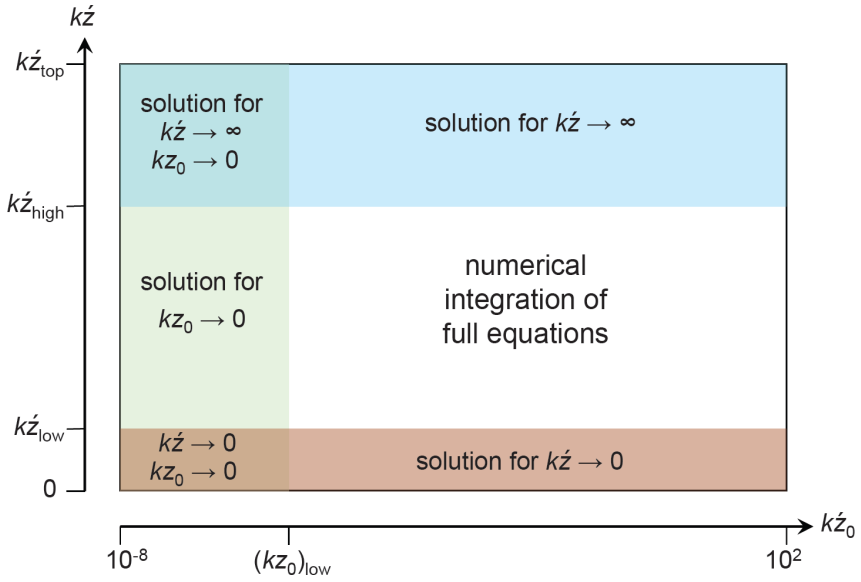


Figure 40: Quilting the solution in $(k\acute{z}, kz_0)$ parameter space.

7.3 Sensitivity to small-scale terrain features

The irregular fluctuations visible on the wind speed profiles calculated by RAMSIM at 10 m a.g.l. over Askervein hill are also seen on those calculated by the linear MSFD model of Beljaars et al. (Figure 37), as well as those yielded by other linear models, for example Salmon et al. (1988, figs. 6, 10, 11 and 16) using MS3DJH/3, a linear model developed by Taylor, Walmsley and Salmon (1983). Similar features can be detected, but are not nearly as pronounced, on profiles calculated by non-linear CFD models, such as those of Castro et al. (plotted on Figure 37) and of Eidsvik (2005, figs. 6-7).

Interestingly, Beljaars et al. (1987, page 297) comment that their modelled Askervein flow is "sensitive to small slope changes in what, at first glance, appears to be a rather smooth hill without small-scale features." Upon comparing profiles of the Askervein flow from various sources, this appears to be a property common to all linear models.

In addition, however, there is reason to believe that the fluctuations visible on our profiles are exaggerated by our coordinate transformation. The Fourier-domain coordinate transformation functions Λ_1 and Λ_3, defined in Equations (30) to (36) (page 35), are stated explicitly here for the two-dimensional case:

$$\begin{aligned} \Lambda_1(k,\acute{z}) &= -\mathrm{i}k\acute{z}\, e^{-|k|\acute{z}}\, \hat{h}_\mathrm{t}(k) \\ \Lambda_3(k,\acute{z}) &= (1+|k|\,\acute{z})\, e^{-|k|\acute{z}} \hat{h}_\mathrm{t}(k) \end{aligned}$$

and plotted in absolute value on Figure 41.

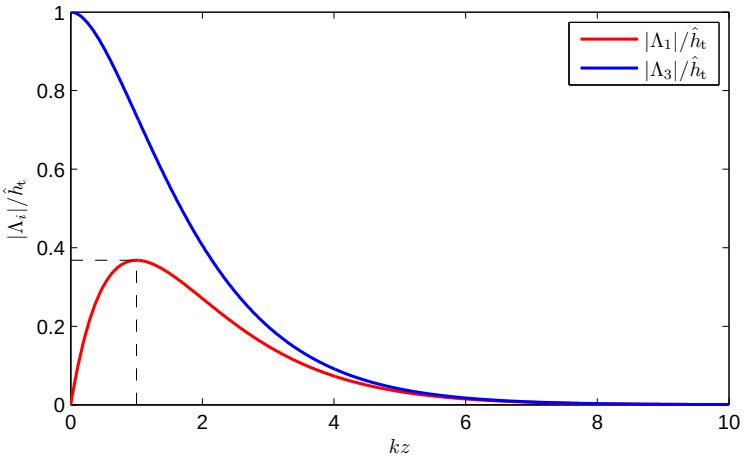


Figure 41: Plot of the absolute value of the coordinate transformation functions Λ_1 and Λ_3, normalized by the terrain height function.

The horizontal transformation Λ_1 has a maximum at $k\acute{z} = 1$. At a given height $\acute{z}$, Λ_1 acts as a blurring filter for high wavenumbers, but as a sharpening filter below $k\acute{z} = 1$, such that the effect of terrain features between, say, $k\acute{z} = 0.5$ and $k\acute{z} = 2$ are much amplified. At $\acute{z} = 10$ m (which roughly corresponds to $h = 10$ m), amplification is exerted on terrain features of length scale $L = 2\pi/k$ between 31 m and 126 m — much larger than $\sigma = 10$ m and thus unaffected by our low-pass filtering of the terrain. In an intuitive sense, these terrain features cause the greatest horizontal compression of vertical "grid lines" at the height considered. No similar sharpening occurs in the vertical coordinate transformation, as Λ_3 acts purely as a blurring filter. The spacing between horizontal grid lines varies monotonically from the ground to the upper boundary. The resulting λ_1 and λ_3 at 10 m a.g.l. over Askervein hill are compared on Figure 42; note the much smoother λ_3.

Upon returning to Cartesian space, the amplification caused by Λ_1 is not completely annulled, due to our linear approximation: the contributions of all the higher-order terms, which in reality should be included for a seamless back-and-forth transition between Cartesian and transformed coordinates, is discarded from the field variables in the transformed coordinate system. The field variables are thus contaminated by residues of the filtering effects of our chosen Λ_1, with greatest "noise" induced from terrain features of the scales mentioned above. This is illustrated on Figure 43: the fluctuations in the Askervein wind speed profile at 10 m a.g.l. correspond exactly to the peaks and troughs in the metric $dx_1/d\acute{x}_1 = 1 + d\lambda_1/d\acute{x}_1$.

Our coordinate transformation thus captures some of the effects of ground-surface curvature on the flow (convex terrain speeds up the flow, concave terrain slows it down), but then exaggerates them.

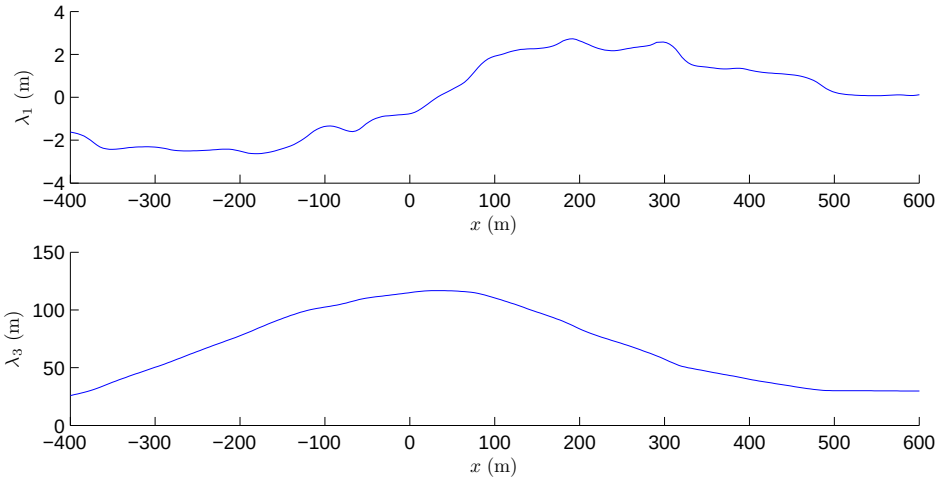


Figure 42: λ_1 and λ_3 at 10 m a.g.l. along Askervein hill line AA.

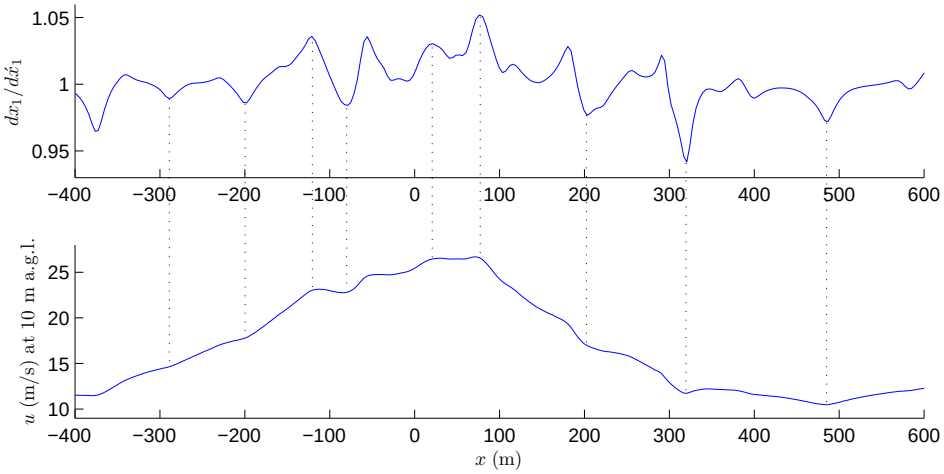


Figure 43: The effect of the coordinate transformation on the speed profile at 10 m a.g.l. along Askervein hill line AA.

Since our current coordinate transformation function introduces these noticeable exaggerations in the flow field, we will devote future efforts to finding a $\Lambda_i(k, \hat{z})$ that has a reduced sharpening effect, while hopefully retaining its more desirable properties.

8 Conclusion

8.1 Development of RAMSIM

The subject of this thesis has been the user-oriented development of a fast and reasonably accurate numerical model for wind flow over orography, designed to be used for wind resource assessment and the siting of wind turbines. The proposed model, RAMSIM, is based on the Reynolds-averaged Navier-Stokes equations for incompressible flow in the neutrally-stratified atmospheric boundary layer, and the $E - \varepsilon$ turbulence closure, all expressed in non-Cartesian coordinates. A terrain-following coordinate system is created from a simple analytical expression that is substituted directly into the governing equations.

Assuming a uniform surface roughness z_0, the equations are linearized by a perturbation expansion about the flat-terrain case. The equations describing the orography-induced perturbation are Fourier-transformed analytically in the two horizontal dimensions, and the pressure and horizontal velocity components are eliminated, resulting in a set of four higher-order ordinary differential equations in $\acute{z}$ (the vertical transformed coordinate). RAMSIM is currently only implemented in two-dimensional space, for which case there are only three ODEs.

The ODEs are solved by numerical integration. Since the 2D equations depend on only two non-dimensional parameters $(k\acute{z}, kz_0)$, intermediate results are saved in a look-up table for all foreseeable combinations of these parameters. The flow perturbation caused by any given orography is then quickly obtained by interpolating the required spectral variables from the LUT, scaling, and returning to physical space by inverse FFT.

8.2 Test cases

Four test cases approximating two-dimensional terrain were examined, allowing us to test our flow model under various conditions:

- A series of hills approximating an infinite sinusoidal (laboratory flow): the simplest possible terrain, consisting of a single wavenumber, and with moderate slopes.
- A single hill with a sinusoidal height profile (laboratory flow): steep slopes.
- Hjardemål escarpment: moderate slope, comparatively abrupt change in slope.
- Askervein hill: large domain, most influence from three-dimensional effects.

The performance of RAMSIM is, on the whole, quite encouraging, though it is affected by some of the classical problems of linearized flow models, namely a poor representation of the flow in the wake region and an overestimated hilltop speed-up effect near the ground. After comparing RAMSIM's calculations over the terrains listed above to those of other models, as well as to measurements from the field and laboratory experiments, one can draw the following conclusions:

- RAMSIM appears to be at least as accurate as WA^{S}P and its cousin LINCOM over upwind slopes and hilltops. While WA^{S}P assumes unchanged streamlines if the flow direction is reversed, and is thus unable to capture any distinction between the upwind and wake regions of a given hill, RAMSIM is able to predict asymmetric flow over symmetric hills, and, to a certain extent, the occurrence of recirculation. Since it calculates the entire flow field at once, RAMSIM is not as fast as WA^{S}P for single-point calculations, but promises to be faster for the calculation of resource grids.

- RAMSIM cannot compete in accuracy with non-linear CFD models; these do a much better job at capturing the wind speed reduction (including recirculation), pressure increase, and TKE increase in the wake region, and do not yield the same near-ground overestimate of the speed-up at hilltops. RAMSIM is, however, orders of magnitude faster, and is therefore suitable for routine use on a personal computer, whereas non-linear CFD models are likely to remain prohibitively expensive in the coming decades, depending on future PC technological improvements.
- It is still doubtful whether RAMSIM yields results much different from those of other linear models such as the MSFD models of Beljaars et al. (1987) and Ayotte and Taylor (1995), though RAMSIM appears to have a slight advantage in certain lee-side wind speed calculations (e.g. Askervein line AA) and in predicting the occurrence of recirculation. RAMSIM is undoubtedly much faster than MSFD models, thanks to the possibility of using results stored in a LUT. Even if no LUT were to be used, RAMSIM would likely remain faster, since its ODEs are solved by direct integration, and no iteration is involved, contrary to MSFD models.

8.3 Future work

Though a version of our flow model has been made functional in two-dimensional space, two main issues must be tackled before RAMSIM can be considered a viable complement or replacement for the WA^{S}P orography model:

Addressing the $kz_0 \to 0$ problem is the first priority; prospective approaches for doing so were discussed in Section 7.2, namely increased machine precision, and quilting the solution in $(kz_0, k\acute{z})$ parameter space using limit solutions.

The second, non-trivial task is to implement the three-dimensional version of the flow model. The equations and boundary conditions have already been derived, and the solution method and overall architecture of the program from the 2D version can be applied with minor modifications. Some work will be required to investigate the effect of the third non-dimensional parameter influencing the equations, namely the angle γ between the forcing applied at the upper boundary and the wavenumber vector. The three-dimensional LUTs required to cover $(kz_0, k\acute{z}, \gamma)$ parameter space may become prohibitively large in terms of storage space and access time. Their resolution may therefore need to be reduced, and other measures taken to maintain the same interpolation accuracy, such as using the information from derivative variables for the interpolation along $k\acute{z}$ (e.g. using $\mathcal{W}', \mathcal{W}'', \mathcal{W}'''$ to interpolate $\mathcal{W}$), or parameterizing each of the the $N_v = 10$ variables in terms of empirical analytical functions of kz_0, $k\acute{z}$ and/or γ.

A further task will be to investigate alternative coordinate transformation formulas that have a lesser sharpening effect, in an effort to reduce the sensitivity of the flow field to small-scale terrain features, as discussed in Section 7.3.

Looking beyond these bare-bones necessities, we will direct our efforts to expanding RAMSIM in the following respects. Firstly, we will investigate the inclusion of some non-linear terms as extra source terms in the first-order equations, which are then solved iteratively, as previously discussed (page 98). Previous authors (Xu et al. 1992, 1994; reviewed by Taylor 1998) attempted this strategy with the MSFD model, and obtained much improved results in the lee of hills, but were simultaneously confronted with intractable numerical stability problems when slopes exceeded 0.3. Since our solution technique is different, there is room for hope that similar difficulties can be avoided.

Secondly, we will work on combining RAMSIM with a model for the effects of roughness changes. In real terrain, z_0 is not uniform: changes in the surface

roughness cause an additional perturbation, not accounted for by RAMSIM. Furthermore, the roughness perturbation can interact with the orography-induced perturbation to produce second-order effects on the flow. As a first-order approximations, however, these second-order interactions can be neglected, and the calculated orography and roughness perturbations simply added together. The current roughness model used by WA^{S}P is a potential candidate; it uses empirical relations to model how roughness changes modify the background flow into a piece-wise logarithmic profile. One can also conceive using an approach similar to RAMSIM's, with ordinary differential equations solved to find the Fourier components of the perturbation, which could then be stored in a LUT.

Thirdly, we will work on relaxing our assumptions on atmospheric boundary-layer flow by re-including some of the effects that are neglected in the current model. The first step will be to include the Coriolis force in our governing equations, along with the effect of finite boundary-layer height, using the Apsley and Castro's (1997) limited-length-scale $E-\varepsilon$ turbulence closure (introduced on page 22). This will imply abandoning the analytical solution for flow over flat terrain discussed in Section 3.3, and resorting to a numerical zero-order solution instead, an example of which was given in Figure 3 (page 24). These curves could then be parameterized using analytical functions. The latitude-dependent Coriolis parameter f_c will become a fourth parameter (in addition to $kz_0, k\acute{z}, \gamma$) on which the equations will depend.

A further step will be the inclusion of a temperature transport equation in order to resolve thermal stratification effects. The temperature field can be coupled to the other flow variables through Apsley and Castro's (1997) limited-length-scale $E-\varepsilon$ closure.

Looking even further ahead, we can envision including more complex effects in our flow model. There has been increasing interest in placing wind turbines in forested areas, not because they offer a favourable wind resource — in fact their very high roughness makes it just the opposite — but simply because they are so abundant. Vegetation canopies offer a special challenge to wind flow modelling enthusiasts, due to the complicated structure of the flow within it, and its wide difference from the free-air flow above. Recent papers have proposed modifications to the $E-\varepsilon$ closure to account for plant drag, notably Foudhil, Brunet and Caltagirone (2005) and Sogachev and Panferov (2006). The incorporation of these modified closures will be investigated, in the view of making a future version of our flow model useable for wind resource assessment in forested areas.

A Derived zero-order quantities

Given the analytical zero-order solution discussed in Section 3.3, here are some derived zero-order quantities that appear in the coefficients of the ordinary differential equations for the first-order variables.

$$
\begin{aligned}
\frac{\partial U_0}{\partial \acute{z}} &= \frac{U_{*0}}{\kappa(\acute{z}+z_0)} \\
\frac{\partial^2 U_0}{\partial \acute{z}^2} &= -\frac{U_{*0}}{\kappa(\acute{z}+z_0)^2} \\
\frac{\partial^3 U_0}{\partial \acute{z}^3} &= 2\frac{U_{*0}}{\kappa(\acute{z}+z_0)^3} \\
\frac{\partial E_0}{\partial \acute{z}} &= 0 \\
\frac{\partial^2 E_0}{\partial \acute{z}^2} &= 0 \\
\frac{\partial \varepsilon_0}{\partial \acute{z}} &= -\frac{U_{*0}^3}{\kappa(\acute{z}+z_0)^2} \\
\frac{1}{\varepsilon_0} &= \frac{\kappa(\acute{z}+z_0)}{U_{*0}^3} \\
\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\right) &= \frac{\kappa}{U_{*0}^3} \\
\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{1}{\varepsilon_0}\right) &= 0 \\
\frac{1}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}} &= -\frac{1}{\acute{z}+z_0} \\
\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) &= \frac{1}{(\acute{z}+z_0)^2} \\
\nu_0\frac{1}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}} &= -\kappa U_{*0} \\
\frac{\partial}{\partial \acute{z}}\left(\nu_0\frac{1}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) &= 0 \\
\nu_0\frac{1}{E_0}\frac{\partial \varepsilon_0}{\partial z} &= -\sqrt{C_\mu}\, U_{*0}^2 \\
\frac{\partial}{\partial \acute{z}}\left(\nu_0\frac{1}{E_0}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) &= 0 \\
\frac{\partial \nu_0}{\partial \acute{z}} &= \kappa U_{*0} \\
\frac{1}{\nu_0}\frac{\partial \nu_0}{\partial \acute{z}} &= \frac{1}{\acute{z}+z_0} \\
\frac{\partial^2 \nu_0}{\partial \acute{z}^2} &= 0
\end{aligned}
$$

$$
\begin{aligned}
\Pi_0 &= \nu_0 \frac{\partial U_0}{\partial \acute{z}} \frac{\partial U_0}{\partial \acute{z}} = \frac{U_{*0}^3}{\kappa(\acute{z}+z_0)} \\
A &= \mathrm{i}k_1 U_0 \\
B &= \mathrm{i}\breve{k}_1 U_0 = -\mathrm{i}k_2 U_0 \\
\frac{\partial A}{\partial \acute{z}} &= \mathrm{i}k_1 \frac{\partial U_0}{\partial \acute{z}} \\
\frac{\partial B}{\partial \acute{z}} &= -\mathrm{i}k_2 \frac{\partial U_0}{\partial \acute{z}} \\
\frac{\partial^2 A}{\partial \acute{z}^2} &= \mathrm{i}k_1 \frac{\partial^2 U_0}{\partial \acute{z}^2} \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{z}}\right) &= -\mathrm{i}k_2 \frac{\partial \sqrt{C_\mu}}{\partial \acute{z}} = 0 \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{z}}\right) &= -\mathrm{i}k_2 \frac{\partial}{\partial \acute{z}}\left(\frac{\kappa(\acute{z}+z_0)}{U_{*0}}\right) = -\mathrm{i}k_2 \frac{\kappa}{U_{*0}} \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right) &= \mathrm{i}k_1 \frac{\partial \sqrt{C_\mu}}{\partial \acute{z}} = 0 \\
\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right) &= \mathrm{i}k_1 \frac{\partial}{\partial \acute{z}}\left(\frac{\kappa(\acute{z}+z_0)}{U_{*0}}\right) = \mathrm{i}k_1 \frac{\kappa}{U_{*0}} \\
\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right) &= \frac{\partial}{\partial \acute{z}}\left(\mathrm{i}k_1 \frac{\kappa}{U_{*0}}\right) = 0
\end{aligned}
$$

B Full derivation of the first-order equations

In Section 4, we gave a summary of how the first-order equations were derived. We present here their full derivation, making use of the shorthand notation introduced on page 50. The continuity equation is examined first, as it is most simple. We then move on to the TKE and dissipation equations; as we will see, their expressions will be required in the momentum equation, due to the term involving the first-order viscosity, ν_1, which is

$$\begin{aligned}
\nu_1 &= \left(C_\mu \frac{E^2}{\varepsilon}\right)_1 \\
&= C_\mu \left(2\frac{E_0 E_1}{\varepsilon_0} - \frac{E_0 E_0}{\varepsilon_0 \varepsilon_0}\varepsilon_1\right) \\
&= C_\mu \frac{(E_0)^2}{\varepsilon_0}\left(2\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0}\right) \\
&= \nu_0 \left(2\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0}\right)
\end{aligned} \tag{B.121}$$

B.1 Continuity equation

We have simply

$$\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_j} = 0 \tag{B.122}$$

Fourier-transforming the continuity equation yields

$$\mathcal{F}\left\{\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_j}\right\} = \mathcal{F}\left\{\frac{\partial \acute{U}_1^{(1)}}{\partial \acute{x}_1} + \frac{\partial \acute{U}_2^{(1)}}{\partial \acute{x}_2} + \frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_3}\right\} = \mathrm{i}k_1 \hat{\acute{U}}_1^{(1)} + \mathrm{i}k_2 \hat{\acute{U}}_2^{(1)} + \frac{\partial \hat{\acute{U}}_3^{(1)}}{\partial \acute{x}_3} = 0$$

which can be rearranged into the simple form

$$\mathrm{i}k_q \hat{\acute{U}}_q^{(1)} = -\mathcal{W}' \tag{B.123}$$

with $q = 1, 2$.

B.2 TKE equation

$$\begin{aligned}
\underbrace{\acute{U}_i \frac{\partial E}{\partial \acute{x}_i}}_{\text{advection}} &= \underbrace{\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_E}\alpha_j^p \frac{\partial}{\partial \acute{x}_q}\left(J\alpha_j^q E\right)\right)}_{\text{diffusion}} \\
&\quad + \underbrace{\frac{\nu}{2J}\left[\frac{\partial}{\partial \acute{x}_p}\left(\left(\alpha_j^p \beta_r^i + \alpha_i^p \beta_r^j\right)\acute{U}_r\right)\right]^2}_{\text{production}} \underbrace{- J\varepsilon}_{\text{dissipation}}
\end{aligned} \tag{B.124}$$

We start by deriving the first-order equation term by term. The advection term:

$$\left[\acute{U}_i \frac{\partial E}{\partial \acute{x}_i}\right]_1 = \acute{U}_i^{(1)} \frac{\partial E_0}{\partial \acute{x}_i} + \acute{U}_i^{(0)} \frac{\partial E_1}{\partial \acute{x}_i} \tag{B.125}$$

The diffusion term:

$$
\begin{aligned}
&\left[\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu}{\sigma_E}\alpha_j^p\frac{\partial}{\partial \acute{x}_q}\Big(J\alpha_j^q E\Big)\right)\right]_1 \\
= \;& \frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_1}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_E}\alpha_{3(1)}^p\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_j}\Big(J_1E_0\Big)\right) \\
&+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha_{j(1)}^q E_0\Big)\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial \acute{x}_j}\right) \\
= \;& \frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\left(2\frac{E_1}{E_0}-\frac{\varepsilon_1}{\varepsilon_0}\right)\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_E}\alpha_{3(1)}^p\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_j}\Big(J_1E_0\Big)\right) \\
&+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha_{j(1)}^q E_0\Big)\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial \acute{x}_j}\right)
\end{aligned}
\tag{B.126}
$$

The production term:

$$
\begin{aligned}
&\frac{\nu}{2J}\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)\acute{U}_r\Big)\right]^2 \\
= \;& \frac{\nu_1}{2}\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)_0\acute{U}_r^{(0)}\Big)\right]^2-\frac{\nu_0J_1}{2}\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)_0\acute{U}_r^{(0)}\Big)\right]^2 \\
&+\nu_0\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)\acute{U}_r\Big)\right]_0\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)\acute{U}_r\Big)\right]_1 \\
= \;& \left(\frac{\nu_0}{2}\Big(2\frac{E_1}{E_0}-\frac{\varepsilon_1}{\varepsilon_0}\Big)-\frac{\nu_0J_1}{2}\right)\left(\frac{\partial\acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)^2 \\
&+\nu_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)_1\acute{U}_r^{(0)}\Big)+\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)_0\acute{U}_r^{(1)}\Big)\right] \\
= \;& \left(2\,\frac{E_1}{E_0}-\frac{\varepsilon_1}{\varepsilon_0}-J_1\right)\frac{\nu_0}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)^2 \\
&+\nu_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(1)}^p\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta_{r(1)}^i\acute{U}_r^{(0)}\big)+\frac{\partial}{\partial \acute{x}_p}\big(\alpha_{i(1)}^p\acute{U}_j^{(0)}\big)\right. \\
&\qquad\left.+\frac{\partial}{\partial \acute{x}_i}\big(\beta_{r(1)}^j\acute{U}_r^{(0)}\big)+\frac{\partial\acute{U}_i^{(1)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(1)}}{\partial \acute{x}_i}\right] \\
= \;& \left(2\,\frac{E_1}{E_0}-\frac{\varepsilon_1}{\varepsilon_0}-J_1\right)\Pi_0 \\
&+2\,\nu_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(1)}^p\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta_{r(1)}^i\acute{U}_r^{(0)}\big)+\frac{\partial\acute{U}_i^{(1)}}{\partial \acute{x}_j}\right]
\end{aligned}
\tag{B.127}
$$

where we used

$$
\begin{aligned}
\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j\big)_0\acute{U}_r^{(\vartheta)}\Big) &= \frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_{j(0)}^p\beta_{r(0)}^i+\alpha_{i(0)}^p\beta_{r(0)}^j\big)\acute{U}_r^{(\vartheta)}\Big) \\
&= \frac{\partial\acute{U}_i^{(\vartheta)}}{\partial \acute{x}_j}+\frac{\partial\acute{U}_j^{(\vartheta)}}{\partial \acute{x}_i}
\end{aligned}
\tag{B.128}
$$

where $\vartheta = 0, 1$ indicates the order of $\acute{U}_r^{(\vartheta)}$. We also used:

$$\begin{aligned}
\frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_j^p\beta_r^i + \alpha_i^p\beta_r^j\big)_1\acute{U}_r^{(\vartheta)}\Big) &= \frac{\partial}{\partial \acute{x}_p}\Big(\big(\alpha_{j(1)}^p\beta_{r(0)}^i + \alpha_{j(0)}^p\beta_{r(1)}^i + \alpha_{i(1)}^p\beta_{r(0)}^j + \alpha_{i(0)}^p\beta_{r(1)}^j\big)\acute{U}_r^{(\vartheta)}\Big) \\
&= \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(1)}^p\beta_{r(0)}^i\acute{U}_r^{(\vartheta)}\big) + \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(0)}^p\beta_{r(1)}^i\acute{U}_r^{(\vartheta)}\big) \\
&\quad + \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{i(1)}^p\beta_{r(0)}^j\acute{U}_r^{(\vartheta)}\big) + \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{i(0)}^p\beta_{r(1)}^j\acute{U}_r^{(\vartheta)}\big) \\
&= \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(1)}^p\acute{U}_i^{(\vartheta)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta_{r(1)}^i\acute{U}_r^{(\vartheta)}\big) \\
&\quad + \frac{\partial}{\partial \acute{x}_p}\big(\alpha_{i(1)}^p\acute{U}_j^{(\vartheta)}\big) + \frac{\partial}{\partial \acute{x}_i}\big(\beta_{r(1)}^j\acute{U}_r^{(\vartheta)}\big) \qquad \text{(B.129)}
\end{aligned}$$

as well as

$$\frac{1}{J} = \frac{1}{1 + s\frac{\partial \lambda_i}{\partial \acute{x}_i} + \ldots} \approx 1 - s\frac{\partial \lambda_i}{\partial \acute{x}_i} - \ldots \qquad \text{(B.130)}$$

hence

$$\Big(\frac{1}{J}\Big)_1 = -\frac{\partial \lambda_i}{\partial \acute{x}_i} = -J_1 \qquad \text{(B.131)}$$

And finally the dissipation term:

$$[-J\varepsilon]_1 = -J_1\varepsilon_0 - \varepsilon_1 \qquad \text{(B.132)}$$

Collecting terms, the first-order TKE equation is:

$$\begin{aligned}
&\frac{\partial(\acute{U}_i^{(1)}E_0)}{\partial \acute{x}_i} + \frac{\partial(\acute{U}_i^{(0)}E_1)}{\partial \acute{x}_i} \\
= \;& \frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_1}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right) + \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_E}\alpha_{3(1)}^p\frac{\partial E_0}{\partial \acute{x}_3}\right) + \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_j}\Big(J_1E_0\Big)\right) \\
&+ \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha_{j(1)}^qE_0\Big)\right) + \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial \acute{x}_j}\right) + \left(2\,\frac{E_1}{E_0} - \frac{\varepsilon_1}{\varepsilon_0} - J_1\right)\Pi_0 \\
&+ 2\,\nu_0\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha_{j(1)}^p\acute{U}_i^{(0)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta_{r(1)}^i\acute{U}_r^{(0)}\big)\right] \\
&+ 2\,\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3} + \frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right) - J_1\varepsilon_0 - \varepsilon_1 \qquad \text{(B.133)}
\end{aligned}$$

Grouping known terms on RHS

$$\begin{aligned}
&\frac{\partial(\acute{U}_i^{(1)}E_0)}{\partial \acute{x}_i} + \frac{\partial(\acute{U}_i^{(0)}E_1)}{\partial \acute{x}_i} - \frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_1}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
&\qquad - \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial \acute{x}_j}\right) - \left(2\,\frac{E_1}{E_0}\right. \\
&\left. - \frac{\varepsilon_1}{\varepsilon_0}\Pi_0 - 2\,\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3} + \frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right) + \varepsilon_1 = S_E \qquad \text{(B.134)}
\end{aligned}$$

where

$$
\begin{aligned}
S_E \;=\;& \frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_E}\alpha^p_{3(1)}\frac{\partial E_0}{\partial \acute{x}_3}\right) + \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_j}\Big(J_1 E_0\Big)\right) \\
& + \frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha^q_{j(1)} E_0\Big)\right) \\
& + 2\,\nu_0\left(\frac{\partial \acute{U}^{(0)}_i}{\partial \acute{x}_j} + \frac{\partial \acute{U}^{(0)}_j}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}^{(0)}_i\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{r(1)}\acute{U}^{(0)}_r\big)\right] \\
& - J_1(\Pi_0 + \varepsilon_0)
\end{aligned}
\tag{B.135}
$$

Fourier transformation

We Fourier-transform the first-order TKE equation term by term:

$$\begin{aligned}\mathcal{F}\left\{\frac{\partial(\acute{U}_i^{(1)}E_0)}{\partial\acute{x}_i}\right\} &= \mathcal{F}\left\{\frac{\partial(\acute{U}_1^{(1)}E_0)}{\partial\acute{x}_1}+\frac{\partial(\acute{U}_2^{(1)}E_0)}{\partial\acute{x}_2}+\frac{\partial(\acute{U}_3^{(1)}E_0)}{\partial\acute{x}_3}\right\}\\ &= \mathrm{i}\left(k_1\hat{\acute{U}}_1^{(1)}+k_2\hat{\acute{U}}_2^{(1)}\right)E_0+\frac{\partial(\hat{\acute{U}}_3^{(1)}E_0)}{\partial\acute{x}_3}\end{aligned}$$

$$\begin{aligned}\mathcal{F}\left\{\frac{\partial(\acute{U}_i^{(0)}E_1)}{\partial\acute{x}_i}\right\} &= \mathcal{F}\left\{\frac{\partial(\acute{U}_1^{(0)}E_1)}{\partial\acute{x}_1}+\frac{\partial(\acute{U}_2^{(0)}E_1)}{\partial\acute{x}_2}\right\}\\ &= \mathrm{i}k_1\acute{U}_1^{(0)}\hat{E}_1+\mathrm{i}k_2\acute{U}_2^{(0)}\hat{E}_1\end{aligned}$$

$$\mathcal{F}\left\{-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_1}{\sigma_E}\frac{\partial E_0}{\partial\acute{x}_3}\right)\right\} = -\frac{\partial}{\partial\acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_E}\frac{\partial E_0}{\partial\acute{x}_3}\right)$$

$$\begin{aligned}\mathcal{F}\left\{-\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial\acute{x}_j}\right)\right\} &= \mathcal{F}\left\{-\frac{\partial}{\partial\acute{x}_1}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial\acute{x}_1}\right)-\frac{\partial}{\partial\acute{x}_2}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial\acute{x}_2}\right)-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial E_1}{\partial\acute{x}_3}\right)\right\}\\ &= (k_1k_1+k_2k_2)\frac{\nu_0}{\sigma_E}\hat{E}_1-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial\hat{E}_1}{\partial\acute{x}_3}\right)\end{aligned}$$

$$\mathcal{F}\left\{-\left(2\,\frac{E_1}{E_0}-\frac{\varepsilon_1}{\varepsilon_0}\right)\Pi_0\right\} = -\left(2\,\frac{\hat{E}_1}{E_0}-\frac{\hat{\varepsilon}_1}{\varepsilon_0}\right)\Pi_0$$

$$\begin{aligned}\mathcal{F}\left\{-2\,\nu_0\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_i}\right)\right\} &= \mathcal{F}\left\{-2\,\nu_0\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_i}\right)\right\}\\ &= -2\,\nu_0\mathcal{F}\left\{\frac{\partial\acute{U}_1^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_1^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_1}\right)+\frac{\partial\acute{U}_2^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_2^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_2}\right)\right\}\\ &= -2\,\nu_0\left[\frac{\partial\acute{U}_1^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\hat{\acute{U}}_1^{(1)}}{\partial\acute{x}_3}+\mathrm{i}k_1\acute{U}_3^{(1)}\right)+\frac{\partial\acute{U}_2^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\hat{\acute{U}}_2^{(1)}}{\partial\acute{x}_3}+\mathrm{i}k_2\acute{U}_3^{(1)}\right)\right]\end{aligned}$$

$$\mathcal{F}\{\varepsilon_1\} = \hat{\varepsilon}_1$$

Collecting terms, the Fourier-transformed TKE equation is

$$\begin{aligned}&\mathrm{i}k_q\acute{U}_q^{(1)}E_0+\frac{\partial(\hat{\acute{U}}_3^{(1)}E_0)}{\partial\acute{x}_3}+\mathrm{i}k_q\acute{U}_q^{(0)}\hat{E}_1-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_E}\frac{\partial E_0}{\partial\acute{x}_3}\right)+k_qk_q\frac{\nu_0}{\sigma_E}\hat{E}_1\\ &-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial\hat{E}_1}{\partial\acute{x}_3}\right)-\left(2\,\frac{\hat{E}_1}{E_0}-\frac{\hat{\varepsilon}_1}{\varepsilon_0}\right)\Pi_0-2\,\nu_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\hat{\acute{U}}_q^{(1)}}{\partial\acute{x}_3}+\mathrm{i}k_q\acute{U}_3^{(1)}\right)+\hat{\varepsilon}_1 = \hat{S}_E\end{aligned}$$

with $q = 1, 2$.

We now Fourier-transform the source term:

$$
\begin{aligned}
& \mathcal{F}\left\{ \frac{\partial}{\partial \acute{x}_p} \left(\frac{\nu_0}{\sigma_E} \alpha^p_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} \right) \right\} \\
= \;& \mathsf{i} k_1 \frac{\nu_0}{\sigma_E} \hat{\alpha}^1_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} + \mathsf{i} k_2 \frac{\nu_0}{\sigma_E} \hat{\alpha}^2_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} + \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \hat{\alpha}^3_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} \right) \\
= \;& \mathsf{i} k_q \frac{\nu_0}{\sigma_E} \hat{\alpha}^q_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} + \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \hat{\alpha}^3_{3(1)} \frac{\partial E_0}{\partial \acute{x}_3} \right)
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{ \frac{\partial}{\partial \acute{x}_j} \left(\frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_j} (J_1 E_0) \right) \right\} \\
= \;& -k_q k_q \frac{\nu_0}{\sigma_E} J_1 E_0 + \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_3} (\hat{J}_1 E_0) \right)
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{ \frac{\partial}{\partial \acute{x}_j} \left(\frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_q} \left(\alpha^q_{j(1)} E_0 \right) \right) \right\} \\
= \;& -k_1 k_1 \frac{\nu_0}{\sigma_E} \hat{\alpha}^1_{1(1)} E_0 - k_1 k_2 \frac{\nu_0}{\sigma_E} \hat{\alpha}^2_{1(1)} E_0 + \mathsf{i} k_1 \frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_3} \left(\hat{\alpha}^3_{1(1)} E_0 \right) \\
& -k_2 k_1 \frac{\nu_0}{\sigma_E} \hat{\alpha}^1_{2(1)} E_0 - k_2 k_2 \frac{\nu_0}{\sigma_E} \hat{\alpha}^2_{2(1)} E_0 + \mathsf{i} k_2 \frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_3} \left(\alpha^3_{2(1)} E_0 \right) \\
& + \mathsf{i} k_1 \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \alpha^1_{3(1)} E_0 \right) + \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_2} \left(\alpha^2_{3(1)} E_0 \right) \right) + \frac{\partial}{\partial \acute{x}_3} \left(\frac{\nu_0}{\sigma_E} \frac{\partial}{\partial \acute{x}_3} \left(\alpha^3_{3(1)} E_0 \right) \right)
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{ 2\,\nu_0 \frac{\partial \acute{U}^{(0)}_i}{\partial \acute{x}_j} \left[\frac{\partial}{\partial \acute{x}_p} (\alpha^p_{j(1)} \acute{U}^{(0)}_i) + \frac{\partial}{\partial \acute{x}_j} (\beta^i_{r(1)} \acute{U}^{(0)}_r) \right] \right\} \\
= \;& 2\,\nu_0 \frac{\partial \acute{U}^{(0)}_q}{\partial \acute{x}_3} \mathcal{F}\Bigg\{ \left[\frac{\partial}{\partial \acute{x}_1} (\alpha^1_{3(1)} \acute{U}^{(0)}_q) + \frac{\partial}{\partial \acute{x}_3} (\beta^q_{1(1)} \acute{U}^{(0)}_1) \right] \\
& \quad + \left[\frac{\partial}{\partial \acute{x}_2} (\alpha^2_{3(1)} \acute{U}^{(0)}_q) + \frac{\partial}{\partial \acute{x}_3} (\beta^q_{2(1)} \acute{U}^{(0)}_2) \right] + \left[\frac{\partial}{\partial \acute{x}_3} (\alpha^3_{3(1)} \acute{U}^{(0)}_q) + \frac{\partial}{\partial \acute{x}_3} (\beta^q_{3(1)} \acute{U}^{(0)}_3) \right] \Bigg\} \\
= \;& 2\,\nu_0 \frac{\partial \acute{U}^{(0)}_q}{\partial \acute{x}_3} \left[\mathsf{i} k_1 \hat{\alpha}^1_{3(1)} \acute{U}^{(0)}_q + \frac{\partial}{\partial \acute{x}_3} (\hat{\beta}^q_{1(1)} \acute{U}^{(0)}_1) + \mathsf{i} k_2 \hat{\alpha}^2_{3(1)} \acute{U}^{(0)}_q + \frac{\partial}{\partial \acute{x}_3} (\hat{\beta}^q_{2(1)} \acute{U}^{(0)}_2) + \frac{\partial}{\partial \acute{x}_3} (\hat{\alpha}^3_{3(1)} \acute{U}^{(0)}_q) \right] \\
= \;& 2\,\nu_0 \frac{\partial \acute{U}^{(0)}_q}{\partial \acute{x}_3} \left[\mathsf{i} k_r \hat{\alpha}^r_{3(1)} \acute{U}^{(0)}_q + \frac{\partial}{\partial \acute{x}_3} (\hat{\alpha}^3_{3(1)} \acute{U}^{(0)}_q) + \frac{\partial}{\partial \acute{x}_3} (\hat{\beta}^q_{r(1)} \acute{U}^{(0)}_r) \right]
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{2\,\nu_0 \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\left[\frac{\partial}{\partial \acute{x}_p}\left(\alpha^p_{j(1)}\acute{U}_i^{(0)}\right)+\frac{\partial}{\partial \acute{x}_j}\left(\beta^i_{r(1)}\acute{U}_r^{(0)}\right)\right]\right\} \\
= \;& 2\,\nu_0\mathcal{F}\left\{\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial}{\partial \acute{x}_j}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)\right)\right\} \\
= \;& 2\,\nu_0\mathcal{F}\left\{\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial}{\partial \acute{x}_1}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)\right)+\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial}{\partial \acute{x}_2}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)\right)\right\} \\
= \;& 2\,\nu_0\left(\mathrm{i}k_1\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_3}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)+\mathrm{i}k_2\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_3}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)\right) \\
= \;& 2\,\mathrm{i}k_q\nu_0\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right)
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{F}\{-J_1\Pi_0\} &= -\hat{J}_1\Pi_0 \\
\mathcal{F}\{-J_1\varepsilon_0\} &= -\hat{J}_1\varepsilon_0
\end{aligned}
$$

Collecting terms, the Fourier-transformed source term is:

$$
\begin{aligned}
\hat{S}_E = \;& \mathrm{i}k_q\frac{\nu_0}{\sigma_E}\hat{\alpha}^q_{3(1)}\frac{\partial E_0}{\partial \acute{x}_3}+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\hat{\alpha}^3_{3(1)}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
& -k_qk_q\frac{\nu_0}{\sigma_E}J_1E_0+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_3}\left(\hat{J}_1E_0\right)\right) \\
& -k_1k_1\frac{\nu_0}{\sigma_E}\hat{\alpha}^1_{1(1)}E_0-k_1k_2\frac{\nu_0}{\sigma_E}\hat{\alpha}^2_{1(1)}E_0+\mathrm{i}k_1\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{1(1)}E_0\right) \\
& -k_2k_1\frac{\nu_0}{\sigma_E}\hat{\alpha}^1_{2(1)}E_0-k_2k_2\frac{\nu_0}{\sigma_E}\hat{\alpha}^2_{2(1)}E_0+\mathrm{i}k_2\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{2(1)}E_0\right) \\
& +\mathrm{i}k_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\alpha^1_{3(1)}E_0\right)+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{3(1)}E_0\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_E}\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{3(1)}E_0\right)\right) \\
& +2\,\nu_0\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\left[\mathrm{i}k_r\hat{\alpha}^r_{3(1)}\acute{U}_q^{(0)}+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_q^{(0)}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^q_{r(1)}\acute{U}_r^{(0)}\right)\right] \\
& +2\,\mathrm{i}k_q\nu_0\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\left(\beta^3_{r(1)}\acute{U}_r^{(0)}\right) \\
& -\hat{J}_1(\Pi_0+\varepsilon_0)
\end{aligned}
\tag{B.136}
$$

Shorthand notation

Rewriting the TKE equation using the continuity equation (81) and the shorthand notation and noting that

$$
\begin{aligned}
& -\frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
= \;& -2\,\frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{E}_1}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{\varepsilon}_1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
= \;& -2\,\hat{E}_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right)+\hat{\varepsilon}_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right) \\
& -2\,\frac{\partial \hat{E}_1}{\partial \acute{x}_3}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right)+\frac{\partial \hat{\varepsilon}_1}{\partial \acute{x}_3}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{x}_3}\right)
\end{aligned}
\tag{B.137}
$$

and that (as will be used in one of the production terms)

$$\begin{aligned}
\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\frac{\partial \hat{\acute{U}}_q^{(1)}}{\partial \acute{x}_3} &= \frac{k_p k_q}{k_r k_r}\frac{\partial \acute{U}_p^{(0)}}{\partial \acute{x}_3}\frac{\partial \hat{\acute{U}}_q^{(1)}}{\partial \acute{x}_3} + \frac{\breve{k}_p \breve{k}_q}{k_r k_r}\frac{\partial \acute{U}_p^{(0)}}{\partial \acute{x}_3}\frac{\partial \hat{\acute{U}}_q^{(1)}}{\partial \acute{x}_3} \\
&= -\frac{1}{k_r k_r}\left(\frac{\partial(\mathrm{i}k_p \acute{U}_p^{(0)})}{\partial \acute{x}_3}\frac{\partial(\mathrm{i}k_q \hat{\acute{U}}_q^{(1)})}{\partial \acute{x}_3} + \frac{\partial(\mathrm{i}\breve{k}_p \acute{U}_p^{(0)})}{\partial \acute{x}_3}\frac{\partial(\mathrm{i}\breve{k}_q \hat{\acute{U}}_q^{(1)})}{\partial \acute{x}_3}\right) \\
&= -\frac{1}{k_r k_r}\left(-\frac{\partial A}{\partial \acute{x}_3}\frac{\partial \mathcal{W}'}{\partial \acute{x}_3} + \frac{\partial B}{\partial \acute{x}_3}\frac{\partial \Omega}{\partial \acute{x}_3}\right) \\
&= \frac{1}{k_r k_r}\left(\frac{\partial A}{\partial \acute{x}_3}\mathcal{W}'' - \frac{\partial B}{\partial \acute{x}_3}\Omega'\right) \qquad \text{(B.138)}
\end{aligned}$$

we have

$$\begin{aligned}
&-\mathcal{W}' E_0 + \mathcal{W}\frac{\partial E_0}{\partial \acute{z}} + \mathcal{W}' E_0 + A\mathcal{E} \\
&-\mathcal{E}\frac{\partial}{\partial \acute{z}}\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) - \mathcal{E}'\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) \\
&+\mathcal{D}\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) + \mathcal{D}'\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) \\
&+k_q k_q \frac{\nu_0}{\sigma_E}\mathcal{E} - \frac{\nu_0}{\sigma_E}\mathcal{E}'' - \frac{1}{\sigma_E}\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{E}' \\
&-2\,\mathcal{E}\frac{\Pi_0}{E_0} + \mathcal{D}\frac{\Pi_0}{\varepsilon_0} - 2\,\nu_0\frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{x}_3}\mathcal{W}'' \\
&+2\,\nu_0\frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{x}_3}\Omega' - 2\,\nu_0\frac{\partial A}{\partial \acute{x}_3}\mathcal{W} + \mathcal{D} \;=\; \hat{S}_E \qquad \text{(B.139)}
\end{aligned}$$

with $q = 1, 2$, i.e.

$$\begin{aligned}
\frac{\partial \mathcal{E}'}{\partial \acute{z}} = \frac{\sigma_E}{\nu_0}\Bigg[& \\
&-2\,\nu_0\frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{z}}\mathcal{W}'' \\
&+\left[\frac{\partial E_0}{\partial \acute{z}} - 2\,\nu_0\frac{\partial A}{\partial \acute{z}}\right]\mathcal{W} \\
&+\left[-\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) - \frac{1}{\sigma_E}\frac{\partial \nu_0}{\partial \acute{z}}\right]\mathcal{E}' \\
&+\left[A - \frac{\partial}{\partial \acute{z}}\left(\frac{2}{E_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) + k_q k_q \frac{\nu_0}{\sigma_E} - 2\,\frac{\Pi_0}{E_0}\right]\mathcal{E} \\
&+\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right)\mathcal{D}' \\
&+\left[\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_E}\frac{\partial E_0}{\partial \acute{z}}\right) + \frac{\Pi_0}{\varepsilon_0} + 1\right]\mathcal{D} \\
&+2\,\nu_0\frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{z}}\Omega' \\
&-\hat{S}_E\Bigg] \qquad \text{(B.140)}
\end{aligned}$$

with $q = 1, 2$. This last equation is written in the form of an ODE, in which we highlight each perturbation variable and its coefficient.

B.3 Dissipation equation

Even though we use the standard $E-\varepsilon$ closure, here we derive the first-order dissipation equation of the limited length-scale $E-\varepsilon$ closure, which we will use in future implementations of our flow model. The standard $E-\varepsilon$ equation is a simplification of this case.

$$\begin{aligned}
\underbrace{\acute{U}_i\frac{\partial\varepsilon}{\partial\acute{x}_i}}_{\text{advection}} &= \underbrace{\frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu}{\sigma_\varepsilon}\alpha_j^p\frac{\partial}{\partial\acute{x}_q}\Big(J\alpha_j^q\varepsilon\Big)\right)}_{\text{diffusion}} \\
&\quad \underbrace{+C_{\varepsilon1}C_\mu\frac{E}{2J}\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)\acute{U}_r\Big)\right]^2}_{\text{production}} \\
&\quad \underbrace{-JC_{\varepsilon2}\frac{\varepsilon^2}{E}}_{\text{destruction}} \qquad\qquad (B.141)
\end{aligned}$$

The advection term:

$$\left[\acute{U}_i\frac{\partial\varepsilon}{\partial\acute{x}_i}\right]_1 = \acute{U}_i^{(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_i}+\acute{U}_i^{(0)}\frac{\partial\varepsilon_1}{\partial\acute{x}_i}$$

The diffusion term:

$$\begin{aligned}
&\left[\frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu}{\sigma_\varepsilon}\alpha_j^p\frac{\partial}{\partial\acute{x}_q}\Big(J\alpha_j^q\varepsilon\Big)\right)\right]_1 \\
= \;& \frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)+\frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu_0}{\sigma_\varepsilon}\alpha_{3(1)}^p\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)\Big)+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_j}\Big(J_1\varepsilon_0\Big)\right) \\
&+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_q}\Big(\alpha_{j(1)}^q\varepsilon_0\Big)\right)+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\varepsilon_1}{\partial\acute{x}_j}\right)
\end{aligned}$$

The production of dissipation term:

$$\begin{aligned}
&\left\{C_{\varepsilon1}C_\mu\frac{E}{2J}\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)\acute{U}_r\Big)\right]^2\right\}_1 \\
= \;& C_{\varepsilon1}C_\mu\frac{E_1}{2}\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)_0\acute{U}_r^{(0)}\Big)\right]^2 \\
&-C_{\varepsilon1}C_\mu\frac{E_0J_1}{2}\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)_0\acute{U}_r^{(0)}\Big)\right]^2 \\
&+C_{\varepsilon1}C_\mu\frac{E_0}{2}2\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)_0\acute{U}_r^{(0)}\Big)\right]\left[\frac{\partial}{\partial\acute{x}_p}\Big((\alpha_j^p\beta_r^i+\alpha_i^p\beta_r^j)\acute{U}_r\Big)\right]_1 \\
= \;& C_{\varepsilon1}C_\mu\frac{E_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2-C_{\varepsilon1}C_\mu\frac{E_0J_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2 \\
&+C_{\varepsilon1}C_\mu E_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)2\left[\frac{\partial}{\partial\acute{x}_p}\Big(\alpha_{j(1)}^p\acute{U}_i^{(0)}\Big)+\frac{\partial}{\partial\acute{x}_j}\Big(\beta_{r(1)}^i\acute{U}_r^{(0)}\Big)+\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_j}\right]
\end{aligned}$$

where we used the previously derived results (B.128), (B.129) and (B.131).

But to minimize the number of derivatives to be calculated, we can write

$$\begin{aligned}\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j}\right] &= \left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}\right)\left[\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j}+\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\right] \\ &= \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3}+\frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right)\end{aligned}$$

The destruction of dissipation term:

$$\begin{aligned}\left[-JC_{\varepsilon 2}\frac{\varepsilon^2}{E}\right]_1 &= -J_1C_{\varepsilon 2}\frac{(\varepsilon_0)^2}{E_0}-C_{\varepsilon 2}\frac{2\varepsilon_0\varepsilon_1}{E_0}+C_{\varepsilon 2}(\varepsilon_0)^2\frac{E_1}{(E_0)^2} \\ &= +C_{\varepsilon 2}\left(-J_1\frac{(\varepsilon_0)^2}{E_0}-\frac{2\varepsilon_0\varepsilon_1}{E_0}+(\varepsilon_0)^2\frac{E_1}{(E_0)^2}\right)\end{aligned}$$

where we used

$$\begin{aligned}\frac{1}{E} &= \frac{1}{E_0+sE_1+\ldots} \\ &= \frac{1}{E_0}\frac{1}{1+s(E_1/E_0)+\ldots} \\ &\approx \frac{1}{E_0}\left(1-s\frac{E_1}{E_0}+\ldots\right) \\ &= \frac{1}{E_0}-s\frac{E_1}{(E_0)^2}+\ldots\end{aligned}$$

hence

$$\left(\frac{1}{E}\right)_1 = -\frac{E_1}{(E_0)^2}$$

Collecting terms, the first-order dissipation equation is:

$$\begin{aligned}&\frac{\partial(\acute{U}_j^{(1)}\varepsilon_0)}{\partial \acute{x}_j}+\frac{\partial(\acute{U}_j^{(0)}\varepsilon_1)}{\partial \acute{x}_j} \\ = \;& \frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_1}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_p}\left(\frac{\nu_0}{\sigma_\varepsilon}\alpha^p_{3(1)}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_j}\Big(J_1\varepsilon_0\Big)\right) \\ &+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_q}\Big(\alpha^q_{j(1)}\varepsilon_0\Big)\right)+\frac{\partial}{\partial \acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_1}{\partial \acute{x}_j}\right) \\ &+C_{\varepsilon 1}C_\mu\frac{E_1}{2}\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)^2+-C_{\varepsilon 1}C_\mu\frac{E_0J_1}{2}\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)^2 \\ &+2\,C_{\varepsilon 1}C_\mu E_0\left(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}+\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{r(1)}\acute{U}_r^{(0)}\big)\right] \\ &+2\,C_{\varepsilon 1}C_\mu E_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\left(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3}+\frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right) \\ &+C_{\varepsilon 2}\left(-J_1\frac{(\varepsilon_0)^2}{E_0}-\frac{2\varepsilon_0\varepsilon_1}{E_0}+(\varepsilon_0)^2\frac{E_1}{(E_0)^2}\right)\end{aligned}$$

Grouping known terms on the RHS

$$\frac{\partial(\acute{U}_j^{(1)}\varepsilon_0)}{\partial\acute{x}_j}+\frac{\partial(\acute{U}_j^{(0)}\varepsilon_1)}{\partial\acute{x}_j}-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)-\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\varepsilon_1}{\partial\acute{x}_j}\right)$$

$$-C_{\varepsilon1}C_\mu\frac{E_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2-2\,C_{\varepsilon1}C_\mu E_0\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_i}\right)$$

$$+C_{\varepsilon2}\left(\frac{2\varepsilon_0\varepsilon_1}{E_0}-(\varepsilon_0)^2\frac{E_1}{(E_0)^2}\right) \;=\; S_\varepsilon$$

where the source term is

$$\begin{aligned}S_\varepsilon \;=\;& \frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu_0}{\sigma_\varepsilon}\alpha^p_{3(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_j}\Big(J_1\varepsilon_0\Big)\right)\\&+\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_q}\Big(\alpha^q_{j(1)}\varepsilon_0\Big)\right)-C_{\varepsilon1}C_\mu\frac{E_0J_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2\\&+2\,C_{\varepsilon1}C_\mu E_0\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)\left[\frac{\partial}{\partial\acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial\acute{x}_j}\big(\beta^i_{r(1)}\acute{U}_r^{(0)}\big)\right]\\&-C_{\varepsilon2}J_1\frac{(\varepsilon_0)^2}{E_0}\end{aligned}$$

Fourier transformation

We Fourier-transform the TKE dissipation equation term by term:

$$\mathcal{F}\left\{\frac{\partial(\acute{U}_j^{(1)}\varepsilon_0)}{\partial\acute{x}_j}\right\} = \mathcal{F}\left\{\frac{\partial(\acute{U}_1^{(1)}\varepsilon_0)}{\partial\acute{x}_1}+\frac{\partial(\acute{U}_2^{(1)}\varepsilon_0)}{\partial\acute{x}_2}+\frac{\partial(\acute{U}_3^{(1)}\varepsilon_0)}{\partial\acute{x}_3}\right\}$$
$$= \mathrm{i}k_q\hat{\acute{U}}_q^{(1)}\varepsilon_0+\frac{\partial(\hat{\acute{U}}_3^{(1)}\varepsilon_0)}{\partial\acute{x}_3}$$

$$\mathcal{F}\left\{\frac{\partial(\acute{U}_j^{(0)}\varepsilon_1)}{\partial\acute{x}_j}\right\} = \mathcal{F}\left\{\frac{\partial(\acute{U}_1^{(0)}\varepsilon_1)}{\partial\acute{x}_1}+\frac{\partial(\acute{U}_2^{(0)}\varepsilon_1)}{\partial\acute{x}_2}\right\}$$
$$= \mathrm{i}k_q\acute{U}_q^{(0)}\hat{\varepsilon}_1$$

$$\mathcal{F}\left\{-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)\right\} = -\frac{\partial}{\partial\acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)$$

$$\mathcal{F}\left\{-\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\varepsilon_1}{\partial\acute{x}_j}\right)\right\} = k_qk_q\frac{\nu_0}{\sigma_\varepsilon}\hat{\varepsilon}_1-\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\hat{\varepsilon}_1}{\partial\acute{x}_3}\right)$$

$$\mathcal{F}\left\{-C_{\varepsilon1}C_\mu\frac{E_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j}+\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2\right\} = \mathcal{F}\left\{-\frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon1}E_1\Pi_0\right\}$$
$$= -\frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon1}\hat{E}_1\Pi_0$$

$$\mathcal{F}\left\{-2\,C_{\varepsilon1}C_\mu E_0\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\acute{U}_i^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_i}\right)\right\}$$
$$= -2\,C_{\varepsilon1}C_\mu E_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\mathcal{F}\left\{\left(\frac{\partial\acute{U}_q^{(1)}}{\partial\acute{x}_3}+\frac{\partial\acute{U}_3^{(1)}}{\partial\acute{x}_q}\right)\right\}$$
$$= -2\,C_{\varepsilon1}C_\mu E_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\hat{\acute{U}}_q^{(1)}}{\partial\acute{x}_3}+\mathrm{i}k_q\acute{U}_3^{(1)}\right)$$

$$\mathcal{F}\left\{C_{\varepsilon2}\left(\frac{2\varepsilon_0\varepsilon_1}{E_0}-(\varepsilon_0)^2\frac{E_1}{(E_0)^2}\right)\right\} = C_{\varepsilon2}\left(\frac{2\varepsilon_0\hat{\varepsilon}_1}{E_0}-(\varepsilon_0)^2\frac{\hat{E}_1}{(E_0)^2}\right)$$

Collecting terms, the Fourier-transformed dissipation equation is:

$$
\begin{aligned}
&\mathrm{i}k_q\hat{\acute{U}}_q^{(1)}\varepsilon_0 + \frac{\partial(\hat{\acute{U}}_3^{(1)}\varepsilon_0)}{\partial\acute{x}_3} + \mathrm{i}k_q\acute{U}_q^{(0)}\hat{\varepsilon}_1 - \frac{\partial}{\partial\acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_\varepsilon}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right) \\
&\quad + k_q k_q\frac{\nu_0}{\sigma_\varepsilon}\hat{\varepsilon}_1 - \frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial\hat{\varepsilon}_1}{\partial\acute{x}_3}\right) - \frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon 1}\hat{E}_1\Pi_0 \\
&\quad - 2\,C_{\varepsilon 1}C_\mu E_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\left(\frac{\partial\hat{\acute{U}}_q^{(1)}}{\partial\acute{x}_3} + \mathrm{i}k_q\acute{U}_3^{(1)}\right) \\
&\quad + C_{\varepsilon 2}\left(\frac{2\varepsilon_0\hat{\varepsilon}_1}{E_0} - (\varepsilon_0)^2\frac{\hat{E}_1}{(E_0)^2}\right) \;=\; \hat{S}_\varepsilon \qquad \text{(B.142)}
\end{aligned}
$$

Fourier-transforming the source term S_ε:

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial\acute{x}_p}\left(\frac{\nu_0}{\sigma_\varepsilon}\alpha^p_{3(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)\right\} \\
= \;& \mathrm{i}k_q\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^q_{3(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^3_{3(1)}\frac{\partial\varepsilon_0}{\partial\acute{x}_3}\right)
\end{aligned}
$$

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_j}\Big(J_1\varepsilon_0\Big)\right)\right\} \\
= \;& -k_q k_q\frac{\nu_0}{\sigma_\varepsilon}\hat{J}_1\varepsilon_0 + \frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_3}\Big(\hat{J}_1\varepsilon_0\Big)\right)
\end{aligned}
$$

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial\acute{x}_j}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_q}\Big(\alpha^q_{j(1)}\varepsilon_0\Big)\right)\right\} \\
= \;& -k_1k_1\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{1(1)}\varepsilon_0 - k_1k_2\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{1(1)}\varepsilon_0 + \mathrm{i}k_1\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_3}\Big(\hat{\alpha}^3_{1(1)}\varepsilon_0\Big) \\
& -k_2k_1\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{2(1)}\varepsilon_0 - k_2k_2\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{2(1)}\varepsilon_0 + \mathrm{i}k_2\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_3}\Big(\hat{\alpha}^3_{2(1)}\varepsilon_0\Big) \\
& + \mathrm{i}k_1\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{3(1)}\varepsilon_0\right) + \mathrm{i}k_2\frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{3(1)}\varepsilon_0\right) + \frac{\partial}{\partial\acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial\acute{x}_3}\Big(\hat{\alpha}^3_{3(1)}\varepsilon_0\Big)\right)
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{F}\left\{-C_{\varepsilon 1}C_\mu\frac{E_0J_1}{2}\left(\frac{\partial\acute{U}_i^{(0)}}{\partial\acute{x}_j} + \frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_i}\right)^2\right\} &= \mathcal{F}\left\{-\frac{\varepsilon_0}{E_0}C_{\varepsilon 1}J_1\Pi_0\right\} \\
&= -\frac{\varepsilon_0}{E_0}C_{\varepsilon 1}\hat{J}_1\Pi_0
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{r(1)}\acute{U}_r^{(0)}\big)\right]\right\} \\
= \;& 2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\mathcal{F}\left\{\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{3(1)}\acute{U}_q^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\beta^q_{r(1)}\acute{U}_r^{(0)}\big)\right]\right\} \\
= \;& 2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\mathcal{F}\Bigg\{\left[\frac{\partial}{\partial \acute{x}_1}\big(\alpha^1_{3(1)}\acute{U}_q^{(0)}\big)+\frac{\partial}{\partial \acute{x}_2}\big(\alpha^2_{3(1)}\acute{U}_q^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\alpha^3_{3(1)}\acute{U}_q^{(0)}\big)\right. \\
& \left.+\frac{\partial}{\partial \acute{x}_3}\big(\beta^q_{1(1)}\acute{U}_1^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\beta^q_{2(1)}\acute{U}_2^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\beta^q_{3(1)}\acute{U}_3^{(0)}\big)\right]\Bigg\} \\
= \;& 2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\left(\mathrm{i}k_r\hat{\alpha}^r_{3(1)}\acute{U}_q^{(0)}+\frac{\partial}{\partial \acute{x}_3}\big(\hat{\alpha}^3_{3(1)}\acute{U}_q^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\hat{\beta}^q_{r(1)}\acute{U}_r^{(0)}\big)\right)
\end{aligned}
$$

$$
\begin{aligned}
& \mathcal{F}\left\{2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\left[\frac{\partial}{\partial \acute{x}_p}\big(\alpha^p_{j(1)}\acute{U}_i^{(0)}\big)+\frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{r(1)}\acute{U}_r^{(0)}\big)\right]\right\} \\
= \;& 2\,C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\mathcal{F}\left\{\left[\frac{\partial}{\partial \acute{x}_q}\big(\beta^3_{r(1)}\acute{U}_r^{(0)}\big)\right]\right\} \\
= \;& 2\,\mathrm{i}k_q C_{\varepsilon 1}C_\mu E_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\beta^3_{r(1)}\acute{U}_r^{(0)}
\end{aligned}
$$

$$
\mathcal{F}\left\{-C_{\varepsilon 2}J_1\frac{(\varepsilon_0)^2}{E_0}\right\} \;=\; -C_{\varepsilon 2}\hat{J}_1\frac{(\varepsilon_0)^2}{E_0}
$$

Collecting terms, the Fourier-transformed source term $\hat{S}_\varepsilon$ is:

$$
\begin{aligned}
\hat{S}_\varepsilon \;=\; & \mathrm{i}k_q\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^q_{3(1)}\frac{\partial\varepsilon_0}{\partial \acute{x}_3}+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^3_{3(1)}\frac{\partial\varepsilon_0}{\partial \acute{x}_3}\right)-k_qk_q\frac{\nu_0}{\sigma_\varepsilon}\hat{J}_1\varepsilon_0+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_3}\Big(\hat{J}_1\varepsilon_0\Big)\right) \\
& -k_1k_1\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{1(1)}\varepsilon_0-k_1k_2\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{1(1)}\varepsilon_0+\mathrm{i}k_1\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_3}\Big(\hat{\alpha}^3_{1(1)}\varepsilon_0\Big) \\
& -k_2k_1\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{2(1)}\varepsilon_0-k_2k_2\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{2(1)}\varepsilon_0+\mathrm{i}k_2\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_3}\Big(\hat{\alpha}^3_{2(1)}\varepsilon_0\Big) \\
& +\mathrm{i}k_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^1_{3(1)}\varepsilon_0\right)+\mathrm{i}k_2\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\hat{\alpha}^2_{3(1)}\varepsilon_0\right)+\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial}{\partial \acute{x}_3}\Big(\hat{\alpha}^3_{3(1)}\varepsilon_0\Big)\right) \\
& +2\,C_{\varepsilon 1}C_\mu E_0\frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\left(\mathrm{i}k_r\hat{\alpha}^r_{3(1)}\acute{U}_q^{(0)}+\frac{\partial}{\partial \acute{x}_3}\big(\hat{\alpha}^3_{3(1)}\acute{U}_q^{(0)}\big)+\frac{\partial}{\partial \acute{x}_3}\big(\hat{\beta}^q_{r(1)}\acute{U}_r^{(0)}\big)+\mathrm{i}k_q\beta^3_{r(1)}\acute{U}_r^{(0)}\right) \\
& -\frac{\varepsilon_0}{E_0}C_{\varepsilon 1}\hat{J}_1\Pi_0-C_{\varepsilon 2}\hat{J}_1\frac{(\varepsilon_0)^2}{E_0} \qquad \text{(B.143)}
\end{aligned}
$$

Shorthand notation

Rewriting the dissipation equation using the continuity equation (Equation (81)), the shorthand notation and noting that

$$\begin{aligned}
-\frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{\nu}_1}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) &= -2\frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{E}_1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) + \frac{\partial}{\partial \acute{x}_3}\left(\frac{\hat{\varepsilon}_1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) \\
&= -2\,\hat{E}_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) + \hat{\varepsilon}_1\frac{\partial}{\partial \acute{x}_3}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) \\
&\quad -2\frac{\partial \hat{E}_1}{\partial \acute{x}_3}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right) + \frac{\partial \hat{\varepsilon}_1}{\partial \acute{x}_3}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{x}_3}\right)
\end{aligned}$$

we have

$$\begin{aligned}
&-\mathcal{W}'\varepsilon_0 + \mathcal{W}\frac{\partial \varepsilon_0}{\partial \acute{z}} + \mathcal{W}'\varepsilon_0 + A\mathcal{D} \\
&-2\,\mathcal{E}\frac{\partial}{\partial \acute{z}}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) + \mathcal{D}\frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) \\
&-2\,\mathcal{E}'\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) + \mathcal{D}'\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) \\
&+k_q k_q\frac{\nu_0}{\sigma_\varepsilon}\mathcal{D} - \frac{\nu_0}{\sigma_\varepsilon}\mathcal{D}'' - \frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\sigma_\varepsilon}\right)\mathcal{D}' - \frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon 1}\mathcal{E}\Pi_0 \\
&-2\,C_{\varepsilon 1}C_\mu E_0\frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{z}}\mathcal{W}'' + 2\,C_{\varepsilon 1}C_\mu E_0\frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{z}}\Omega' \\
&-2\,C_{\varepsilon 1}C_\mu E_0\frac{\partial A}{\partial \acute{z}}\mathcal{W} + C_{\varepsilon 2}\frac{2\varepsilon_0}{E_0}\mathcal{D} - C_{\varepsilon 2}\left(\frac{\varepsilon_0}{E_0}\right)^2\mathcal{E} = \hat{S}_\varepsilon
\end{aligned}$$

i.e.

$$\begin{aligned}
\frac{\partial \mathcal{D}'}{\partial \acute{z}} = \frac{\sigma_\varepsilon}{\nu_0}\Bigg[& \\
&+\left[\frac{\partial \varepsilon_0}{\partial \acute{z}} - 2\,C_{\varepsilon 1}C_\mu E_0\frac{\partial A}{\partial \acute{z}}\right]\mathcal{W} \\
&-2\,C_{\varepsilon 1}C_\mu E_0\frac{1}{k_r k_r}\frac{\partial A}{\partial \acute{z}}\mathcal{W}'' \\
&-2\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right)\mathcal{E}' \\
&+\left[-2\frac{\partial}{\partial \acute{z}}\left(\frac{1}{E_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) - \frac{\varepsilon_0}{(E_0)^2}C_{\varepsilon 1}\Pi_0 - C_{\varepsilon 2}\left(\frac{\varepsilon_0}{E_0}\right)^2\right]\mathcal{E} \\
&+\frac{1}{\sigma_\varepsilon}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial \varepsilon_0}{\partial \acute{z}} - \frac{\partial \nu_0}{\partial \acute{z}}\right)\mathcal{D}' \\
&+\left[A + \frac{\partial}{\partial \acute{z}}\left(\frac{1}{\varepsilon_0}\frac{\nu_0}{\sigma_\varepsilon}\frac{\partial \varepsilon_0}{\partial \acute{z}}\right) + C_{\varepsilon 2}\frac{2\varepsilon_0}{E_0} + k_q k_q\frac{\nu_0}{\sigma_\varepsilon}\right]\mathcal{D} \\
&+2\,C_{\varepsilon 1}C_\mu E_0\frac{1}{k_r k_r}\frac{\partial B}{\partial \acute{z}}\Omega' \\
&-\hat{S}_\varepsilon\Bigg]
\end{aligned} \tag{B.144}$$

with $q = 1, 2$.

B.4 Momentum transport equation

$$\underbrace{\frac{\partial}{\partial \acute{x}_q}\left(\alpha_j^q \nu \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)\acute{U}_p\Big)\right)}_{\text{diffusion}} - \underbrace{\frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J\alpha_i^q p\right)}_{\text{pressure}} = \underbrace{\acute{U}_j \frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J}\beta_q^i \acute{U}_q\right)}_{\text{advection}} \tag{B.145}$$

The first-order advection term:

$$\begin{aligned}\left[\acute{U}_j \frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J}\beta_q^i \acute{U}_q\right)\right]_1 &= \acute{U}_j^{(1)} \frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J_0}\beta_{q(0)}^i \acute{U}_q^{(0)}\right) + \acute{U}_j^{(0)} \frac{\partial}{\partial \acute{x}_j}\left(\Big(\frac{1}{J}\Big)_1 \beta_{q(0)}^i \acute{U}_q^{(0)}\right) \\ &\quad + \acute{U}_j^{(0)} \frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J_0}\beta_{q(1)}^i \acute{U}_q^{(0)}\right) + \acute{U}_j^{(0)} \frac{\partial}{\partial \acute{x}_j}\left(\frac{1}{J_0}\beta_{q(0)}^i \acute{U}_q^{(1)}\right) \\ &= \acute{U}_j^{(1)} \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \acute{U}_j^{(0)} \frac{\partial}{\partial \acute{x}_j}\left(-J_1 \acute{U}_i^{(0)} + \beta_{q(1)}^i \acute{U}_q^{(0)} + \acute{U}_i^{(1)}\right)\end{aligned} \tag{B.146}$$

where we used $(J^{-1})_1 = -J_1$.

The first-order pressure term:

$$\begin{aligned}\left(-\frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J\alpha_i^q p\right)\right)_1 &= -\frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J_1 \alpha_{i(0)}^q p_0\right) - \frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J_0 \alpha_{i(1)}^q p_0\right) - \frac{1}{\rho}\frac{\partial}{\partial \acute{x}_q}\left(J_0 \alpha_{i(0)}^q p_1\right) \\ &= -\frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_i}\end{aligned} \tag{B.147}$$

In expanding the first-order diffusion term, we use the results derived in Equations (B.128) and (B.129).

The first-order diffusion term:

$$\begin{aligned}&\left[\frac{\partial}{\partial \acute{x}_q}\left(\alpha_j^q \nu \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)\acute{U}_p\Big)\right)\right]_1 \\ = \;& \frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(1)}^q \nu_0 \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)_0 \acute{U}_p^{(0)}\Big)\right) + \frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(0)}^q \nu_1 \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)_0 \acute{U}_p^{(0)}\Big)\right) \\ &+ \frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(0)}^q \nu_0 \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)_1 \acute{U}_p^{(0)}\Big)\right) + \frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(0)}^q \nu_0 \frac{\partial}{\partial \acute{x}_r}\Big(\big(\alpha_j^r\beta_p^i + \alpha_i^r\beta_p^j\big)_0 \acute{U}_p^{(1)}\Big)\right) \\ = \;& \frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(1)}^q \nu_0 \Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) + \frac{\partial}{\partial \acute{x}_j}\left(\nu_1 \Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) \\ &+ \frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \Big(\frac{\partial}{\partial \acute{x}_r}\big(\alpha_{j(1)}^r \acute{U}_i^{(0)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta_{p(1)}^i \acute{U}_p^{(0)}\big) + \frac{\partial}{\partial \acute{x}_r}\big(\alpha_{i(1)}^r \acute{U}_j^{(0)}\big) + \frac{\partial}{\partial \acute{x}_i}\big(\beta_{p(1)}^j \acute{U}_p^{(0)}\big)\Big)\right) \\ &+ \frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \Big(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\Big)\right)\end{aligned} \tag{B.148}$$

Collecting terms, the first-order momentum transport equation is

$$\begin{aligned}&\frac{\partial}{\partial \acute{x}_q}\left(\alpha_{j(1)}^q \nu_0 \Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) + \frac{\partial}{\partial \acute{x}_j}\left(\nu_1 \Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) \\ &+ \frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \Big(\frac{\partial}{\partial \acute{x}_r}\big(\alpha_{j(1)}^r \acute{U}_i^{(0)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta_{p(1)}^i \acute{U}_p^{(0)}\big) + \frac{\partial}{\partial \acute{x}_r}\big(\alpha_{i(1)}^r \acute{U}_j^{(0)}\big) + \frac{\partial}{\partial \acute{x}_i}\big(\beta_{p(1)}^j \acute{U}_p^{(0)}\big)\Big)\right) \\ &+ \frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \Big(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\Big)\right) - \frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_i} \\ = \;& \acute{U}_j^{(1)} \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \acute{U}_j^{(0)} \frac{\partial}{\partial \acute{x}_j}\left(-J_1 \acute{U}_i^{(0)} + \beta_{q(1)}^i \acute{U}_q^{(0)} + \acute{U}_i^{(1)}\right)\end{aligned} \tag{B.149}$$

Grouping known terms on the RHS

We now place all terms containing the unknowns $\acute{U}_i^{(1)}$, ν_1 and p_1 on the LHS, and group the other terms containing only zero-order flow variables (and known first-order geometry factors J_1, $\beta^i_{j(1)}$ and $\alpha^j_{i(1)}$) into the source term S_i.

$$\begin{aligned}\acute{U}_j^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \acute{U}_j^{(0)}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} - \frac{\partial}{\partial \acute{x}_j}\left(\nu_0\Big(\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\Big)\right) \\ -\frac{\partial}{\partial \acute{x}_j}\left(\nu_1\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) + \frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_i} &= S_i\end{aligned}$$

where

$$\begin{aligned}S_i &= \acute{U}_j^{(0)}\frac{\partial}{\partial \acute{x}_j}\Big(J_1\acute{U}_i^{(0)} - \beta^i_{q(1)}\acute{U}_q^{(0)}\Big) + \frac{\partial}{\partial \acute{x}_q}\left(\alpha^q_{j(1)}\nu_0\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right) \\ &+\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\Big(\frac{\partial}{\partial \acute{x}_r}\big(\alpha^r_{j(1)}\acute{U}_i^{(0)}\big) + \frac{\partial}{\partial \acute{x}_j}\big(\beta^i_{p(1)}\acute{U}_p^{(0)}\big) + \frac{\partial}{\partial \acute{x}_r}\big(\alpha^r_{i(1)}\acute{U}_j^{(0)}\big) + \frac{\partial}{\partial \acute{x}_i}\big(\beta^j_{p(1)}\acute{U}_p^{(0)}\big)\Big)\right)\end{aligned}$$

Fourier-transformation of the momentum transport equation

$$\begin{aligned}\mathcal{F}\left\{\acute{U}_j^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}\right\} &= \mathcal{F}\Big\{\acute{U}_1^{(1)}\underbrace{\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_1}}_{=0} + \acute{U}_2^{(1)}\underbrace{\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_2}}_{=0} + \acute{U}_3^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\Big\} \\ &= \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\mathcal{F}\left\{\acute{U}_3^{(1)}\right\} = \hat{\acute{U}}_3^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\end{aligned}$$

In the case $i = 3$, this term is zero since $\acute{U}_3^{(0)} = 0$.

$$\begin{aligned}\mathcal{F}\Big\{\acute{U}_j^{(0)}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j}\Big\} &= \mathcal{F}\Big\{\acute{U}_1^{(0)}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_1} + \acute{U}_2^{(0)}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_2} + \underbrace{\acute{U}_3^{(0)}}_{=0}\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3}\Big\} \\ &= \acute{U}_1^{(0)}\mathcal{F}\Big\{\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_1}\Big\} + \acute{U}_2^{(0)}\mathcal{F}\Big\{\frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_2}\Big\} \\ &= \mathsf{i}k_1\acute{U}_1^{(0)}\hat{\acute{U}}_i^{(1)} + \mathsf{i}k_2\acute{U}_2^{(0)}\hat{\acute{U}}_i^{(1)} \\ &= \mathsf{i}\Big(k_1\acute{U}_1^{(0)} + k_2\acute{U}_2^{(0)}\Big)\hat{\acute{U}}_i^{(1)}\end{aligned}$$

$$\begin{aligned}\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_1\Big(\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j} + \frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\Big)\right)\right\} &= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\nu_1\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_i}\right) + \frac{\partial}{\partial \acute{x}_2}\left(\nu_1\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_i}\right) + \frac{\partial}{\partial \acute{x}_3}\left(\nu_1\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)\right\} \\ &= \mathsf{i}k_1\hat{\nu}_1\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_i} + \mathsf{i}k_2\hat{\nu}_1\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_i} + \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right) \\ &= \mathsf{i}\hat{\nu}_1\left(k_1\frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_i} + k_2\frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_i}\right) + \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)\end{aligned}$$

Note that the first two terms vanish for $i = 1, 2$ and the last term vanishes for $i = 3$.

$$\mathcal{F}\left\{ -\frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_j}\right)\right\} = \mathcal{F}\left\{ -\frac{\partial}{\partial \acute{x}_1}\left(\nu_0 \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_1}\right) - \frac{\partial}{\partial \acute{x}_2}\left(\nu_0 \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_2}\right) - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \acute{U}_i^{(1)}}{\partial \acute{x}_3}\right)\right\}$$

$$= k_1 k_1 \nu_0 \hat{\acute{U}}_i^{(1)} + k_2 k_2 \nu_0 \hat{\acute{U}}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{\acute{U}}_i^{(1)}}{\partial \acute{x}_3}\right)$$

$$= \left(k_1 k_1 + k_2 k_2\right) \nu_0 \hat{\acute{U}}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{\acute{U}}_i^{(1)}}{\partial \acute{x}_3}\right)$$

Looking at the term

$$-\frac{\partial}{\partial \acute{x}_j}\left(\nu_0 \frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\right) = -\frac{\partial \nu_0}{\partial \acute{x}_j}\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i} - \nu_0 \frac{\partial}{\partial \acute{x}_j}\left(\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_i}\right)$$

$$= -\frac{\partial \nu_0}{\partial \acute{x}_3}\frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i} - \nu_0 \frac{\partial}{\partial \acute{x}_i}\underbrace{\left(\frac{\partial \acute{U}_j^{(1)}}{\partial \acute{x}_j}\right)}_{=0}$$

where we made use of the first-order continuity equation (80) to eliminate a term.

In the cases $i = 1, 2$,

$$\mathcal{F}\left\{ -\frac{\partial \nu_0}{\partial \acute{x}_3}\frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right\} = -\mathrm{i}k_i \frac{\partial \nu_0}{\partial \acute{x}_3}\hat{\acute{U}}_3^{(1)}$$

$$\mathcal{F}\left\{ \frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_i}\right\} = \frac{1}{\rho}\mathrm{i}k_i \hat{p}_1$$

In the case $i = 3$,

$$\mathcal{F}\left\{ -\frac{\partial \nu_0}{\partial \acute{x}_3}\frac{\partial \acute{U}_3^{(1)}}{\partial \acute{x}_i}\right\} = -\frac{\partial \nu_0}{\partial \acute{x}_3}\frac{\partial \hat{\acute{U}}_3^{(1)}}{\partial \acute{x}_i}$$

$$\mathcal{F}\left\{ \frac{1}{\rho}\frac{\partial p_1}{\partial \acute{x}_3}\right\} = \frac{1}{\rho}\frac{\partial \hat{p}_1}{\partial \acute{x}_3}$$

Collecting terms, we have, in the case $i = 1, 2$:

$$\hat{\acute{U}}_3^{(1)}\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3} + \mathrm{i}\left(k_1 \acute{U}_1^{(0)} + k_2 \acute{U}_2^{(0)}\right)\hat{\acute{U}}_i^{(1)}$$

$$+ \left(k_1 k_1 + k_2 k_2\right)\nu_0 \hat{\acute{U}}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{\acute{U}}_i^{(1)}}{\partial \acute{x}_3}\right)$$

$$-\mathrm{i}k_i \frac{\partial \nu_0}{\partial \acute{x}_3}\hat{\acute{U}}_3^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1 \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right) + \frac{1}{\rho}\mathrm{i}k_i \hat{p}_1 = \hat{S}_i \qquad \text{(B.150)}$$

and in the case $i = 3$:

$$\mathrm{i}\left(k_1 \acute{U}_1^{(0)} + k_2 \acute{U}_2^{(0)}\right)\hat{\acute{U}}_3^{(1)} + \left(k_1 k_1 + k_2 k_2\right)\nu_0 \hat{\acute{U}}_3^{(1)}$$

$$-\frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{\acute{U}}_3^{(1)}}{\partial \acute{x}_3}\right) - \frac{\partial \nu_0}{\partial \acute{x}_3}\frac{\partial \hat{\acute{U}}_3^{(1)}}{\partial \acute{x}_i}$$

$$-\mathrm{i}\hat{\nu}_1\left(k_1 \frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_3} + k_2 \frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_3}\right) + \frac{1}{\rho}\frac{\partial \hat{p}_1}{\partial \acute{x}_3} = \hat{S}_3 \qquad \text{(B.151)}$$

We now Fourier-transform the source term S_i, term by term:

$$\begin{aligned}
\mathcal{F}\left\{\acute{U}_j^{(0)}\frac{\partial}{\partial \acute{x}_j}\left(J_1\acute{U}_i^{(0)}-\beta^i_{q(1)}\acute{U}_q^{(0)}\right)\right\} &= \mathsf{i}k_1\acute{U}_1^{(0)}\left(\hat{J}_1\acute{U}_i^{(0)}-\hat{\beta}^i_{q(1)}\acute{U}_q^{(0)}\right)+\mathsf{i}k_2\acute{U}_2^{(0)}\left(\hat{J}_1\acute{U}_i^{(0)}-\hat{\beta}^i_{q(1)}\acute{U}_q^{(0)}\right)\\
&= \mathsf{i}\left(k_1\acute{U}_1^{(0)}+k_2\acute{U}_2^{(0)}\right)\left(\hat{J}_1\acute{U}_i^{(0)}-\hat{\beta}^i_{q(1)}\acute{U}_q^{(0)}\right)
\end{aligned}$$

$$\begin{aligned}
\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_q}\left(\alpha^q_{j(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_j}\right)\right\} &= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\alpha^1_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)\right\}\\
&= \mathsf{i}k_1\hat{\alpha}^1_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}+\mathsf{i}k_2\hat{\alpha}^2_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)\\
&= \mathsf{i}\left(k_1\hat{\alpha}^1_{3(1)}+k_2\hat{\alpha}^2_{3(1)}\right)\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right)
\end{aligned}$$

$$\begin{aligned}
\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_q}\left(\alpha^q_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\right\} &= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\alpha^1_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)+\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\right\}\\
&= \mathsf{i}k_1\hat{\alpha}^1_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}+\mathsf{i}k_2\hat{\alpha}^2_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)\\
&= \mathsf{i}\left(k_1\hat{\alpha}^1_{j(1)}+k_2\hat{\alpha}^2_{j(1)}\right)\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}+\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{j(1)}\nu_0\frac{\partial \acute{U}_j^{(0)}}{\partial \acute{x}_i}\right)
\end{aligned}$$

$$\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial}{\partial \acute{x}_r}\left(\alpha^r_{j(1)}\acute{U}_i^{(0)}\right)\right)\right\}\\
=\;&\mathcal{F}\Bigg\{\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_1}\left(\alpha^1_{1(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{1(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{1(1)}\acute{U}_i^{(0)}\right)\right)\\
&+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_1}\left(\alpha^1_{2(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{2(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{2(1)}\acute{U}_i^{(0)}\right)\right)\\
&+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_1}\left(\alpha^1_{3(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_2}\left(\alpha^2_{3(1)}\acute{U}_i^{(0)}\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\alpha^3_{3(1)}\acute{U}_i^{(0)}\right)\right)\Bigg\}\\
=\;&\mathsf{i}k_1\nu_0\mathsf{i}k_1\hat{\alpha}^1_{1(1)}\acute{U}_i^{(0)}+\mathsf{i}k_1\nu_0\mathsf{i}k_2\hat{\alpha}^2_{1(1)}\acute{U}_i^{(0)}+\mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{1(1)}\acute{U}_i^{(0)}\right)\\
&+\mathsf{i}k_2\nu_0\mathsf{i}k_1\hat{\alpha}^1_{2(1)}\acute{U}_i^{(0)}+\mathsf{i}k_2\nu_0\mathsf{i}k_2\hat{\alpha}^2_{2(1)}\acute{U}_i^{(0)}+\mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{2(1)}\acute{U}_i^{(0)}\right)\\
&+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\mathsf{i}k_1\hat{\alpha}^1_{3(1)}\acute{U}_i^{(0)}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\mathsf{i}k_2\hat{\alpha}^2_{3(1)}\acute{U}_i^{(0)}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_i^{(0)}\right)\right)\\
=\;&-k_1k_1\nu_0\hat{\alpha}^1_{1(1)}\acute{U}_i^{(0)}-k_1k_2\nu_0\hat{\alpha}^2_{1(1)}\acute{U}_i^{(0)}+\mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{1(1)}\acute{U}_i^{(0)}\right)\\
&-k_2k_1\nu_0\hat{\alpha}^1_{2(1)}\acute{U}_i^{(0)}-k_2k_2\nu_0\hat{\alpha}^2_{2(1)}\acute{U}_i^{(0)}+\mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{2(1)}\acute{U}_i^{(0)}\right)\\
&+\mathsf{i}k_1\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\alpha}^1_{3(1)}\acute{U}_i^{(0)}\right)+\mathsf{i}k_2\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\alpha}^2_{3(1)}\acute{U}_i^{(0)}\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_i^{(0)}\right)\right)
\end{aligned}$$

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial}{\partial \acute{x}_j}\left(\beta^i_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_1}\left(\beta^i_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_2}\left(\beta^i_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\beta^i_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= -k_1k_1\nu_0\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p - k_2k_2\nu_0\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p + \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p\right)\right)
\end{aligned}
$$

$$
\begin{aligned}
\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial}{\partial \acute{x}_r}\left(\alpha^r_{i(1)}\acute{U}^{(0)}_j\right)\right)\right\} &= -k_1k_1\nu_0\hat{\alpha}^1_{i(1)}\acute{U}^{(0)}_1 - k_1k_2\nu_0\hat{\alpha}^2_{i(1)}\acute{U}^{(0)}_1 + \mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{i(1)}\acute{U}^{(0)}_1\right)\\
&\quad -k_2k_1\nu_0\hat{\alpha}^1_{i(1)}\acute{U}^{(0)}_2 - k_2k_2\nu_0\hat{\alpha}^2_{i(1)}\acute{U}^{(0)}_2 + \mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{i(1)}\acute{U}^{(0)}_2\right)
\end{aligned}
$$

Case $i = 1, 2$:

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial}{\partial \acute{x}_i}\left(\beta^j_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_i}\left(\beta^1_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_i}\left(\beta^2_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_i}\left(\beta^3_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= -k_1k_i\nu_0\hat{\beta}^1_{p(1)}\acute{U}^{(0)}_p - k_2k_i\nu_0\hat{\beta}^2_{p(1)}\acute{U}^{(0)}_p + \mathsf{i}k_i\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\beta}^3_{p(1)}\acute{U}^{(0)}_p\right)
\end{aligned}
$$

Case $i = 3$:

$$
\begin{aligned}
&\mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_j}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\beta^j_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= \mathcal{F}\left\{\frac{\partial}{\partial \acute{x}_1}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\beta^1_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_2}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\beta^2_{p(1)}\acute{U}^{(0)}_p\right)\right)+\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\beta^3_{p(1)}\acute{U}^{(0)}_p\right)\right)\right\}\\
&= \mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^1_{p(1)}\acute{U}^{(0)}_p\right) + \mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^2_{p(1)}\acute{U}^{(0)}_p\right) + \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^3_{p(1)}\acute{U}^{(0)}_p\right)\right)
\end{aligned}
$$

Collecting terms, we have, for $i = 1, 2$,

$$
\begin{aligned}
\hat{S}_i &= \mathsf{i}\left(k_1\acute{U}^{(0)}_1 + k_2\acute{U}^{(0)}_2\right)\left(\hat{J}_1\acute{U}^{(0)}_i - \hat{\beta}^i_{q(1)}\acute{U}^{(0)}_q\right)\\
&\quad +\mathsf{i}\left(k_1\hat{\alpha}^1_{3(1)} + k_2\hat{\alpha}^2_{3(1)}\right)\nu_0\frac{\partial \acute{U}^{(0)}_i}{\partial \acute{x}_3} + \frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial \acute{U}^{(0)}_i}{\partial \acute{x}_3}\right)\\
&\quad -k_1k_1\nu_0\hat{\alpha}^1_{1(1)}\acute{U}^{(0)}_i - k_1k_2\nu_0\hat{\alpha}^2_{1(1)}\acute{U}^{(0)}_i + \mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{1(1)}\acute{U}^{(0)}_i\right)\\
&\quad -k_2k_1\nu_0\hat{\alpha}^1_{2(1)}\acute{U}^{(0)}_i - k_2k_2\nu_0\hat{\alpha}^2_{2(1)}\acute{U}^{(0)}_i + \mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{2(1)}\acute{U}^{(0)}_i\right)\\
&\quad +\mathsf{i}k_1\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\alpha}^1_{3(1)}\acute{U}^{(0)}_i\right) + \mathsf{i}k_2\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\alpha}^2_{3(1)}\acute{U}^{(0)}_i\right) + \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}^{(0)}_i\right)\right)\\
&\quad -k_1k_1\nu_0\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p - k_2k_2\nu_0\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p + \frac{\partial}{\partial \acute{x}_3}\left(\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}^i_{p(1)}\acute{U}^{(0)}_p\right)\right)\\
&\quad -k_1k_1\nu_0\hat{\alpha}^1_{i(1)}\acute{U}^{(0)}_1 - k_1k_2\nu_0\hat{\alpha}^2_{i(1)}\acute{U}^{(0)}_1 + \mathsf{i}k_1\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{i(1)}\acute{U}^{(0)}_1\right)\\
&\quad -k_2k_1\nu_0\hat{\alpha}^1_{i(1)}\acute{U}^{(0)}_2 - k_2k_2\nu_0\hat{\alpha}^2_{i(1)}\acute{U}^{(0)}_2 + \mathsf{i}k_2\nu_0\frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}^3_{i(1)}\acute{U}^{(0)}_2\right)\\
&\quad -k_1k_i\nu_0\hat{\beta}^1_{p(1)}\acute{U}^{(0)}_p - k_2k_i\nu_0\hat{\beta}^2_{p(1)}\acute{U}^{(0)}_p + \mathsf{i}k_i\frac{\partial}{\partial \acute{x}_3}\left(\nu_0\hat{\beta}^3_{p(1)}\acute{U}^{(0)}_p\right) \qquad \text{(B.152)}
\end{aligned}
$$

and, for $i = 3$,

$$\begin{aligned}
\hat{S}_3 \;=\;& -\mathsf{i}\left(k_1\acute{U}_1^{(0)} + k_2\acute{U}_2^{(0)}\right)\hat{\beta}^3_{q(1)}\acute{U}_q^{(0)} \\
& +\mathsf{i}\left(k_1\hat{\alpha}^1_{j(1)} + k_2\hat{\alpha}^2_{j(1)}\right)\nu_0\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{j(1)}\nu_0\frac{\partial\acute{U}_j^{(0)}}{\partial\acute{x}_3}\right) \\
& -k_1k_1\nu_0\hat{\beta}^3_{p(1)}\acute{U}_p^{(0)} - k_2k_2\nu_0\hat{\beta}^3_{p(1)}\acute{U}_p^{(0)} + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{p(1)}\acute{U}_p^{(0)}\right)\right) \\
& -k_1k_1\nu_0\hat{\alpha}^1_{3(1)}\acute{U}_1^{(0)} - k_1k_2\nu_0\hat{\alpha}^2_{3(1)}\acute{U}_1^{(0)} + \mathsf{i}k_1\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_1^{(0)}\right) \\
& -k_2k_1\nu_0\hat{\alpha}^1_{3(1)}\acute{U}_2^{(0)} - k_2k_2\nu_0\hat{\alpha}^2_{3(1)}\acute{U}_2^{(0)} + \mathsf{i}k_2\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_2^{(0)}\right) \\
& +\mathsf{i}k_1\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^1_{p(1)}\acute{U}_p^{(0)}\right) + \mathsf{i}k_2\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^2_{p(1)}\acute{U}_p^{(0)}\right) + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{p(1)}\acute{U}_p^{(0)}\right)\right)
\end{aligned} \tag{B.153}$$

Note that the source term can also be written

$$\begin{aligned}
\hat{S}_q \;=\;& \mathsf{i}k_r\acute{U}_r^{(0)}\left(\hat{J}_1\acute{U}_q^{(0)} - \hat{\beta}^q_{r(1)}\acute{U}_r^{(0)}\right) \\
& +\mathsf{i}k_r\hat{\alpha}^r_{3(1)}\nu_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial\acute{U}_q^{(0)}}{\partial\acute{x}_3}\right) \\
& -k_sk_r\nu_0\hat{\alpha}^r_{s(1)}\acute{U}_q^{(0)} + \mathsf{i}k_s\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{s(1)}\acute{U}_q^{(0)}\right) \\
& +\mathsf{i}k_r\frac{\partial}{\partial\acute{x}_3}\left(\nu_0\hat{\alpha}^r_{3(1)}\acute{U}_q^{(0)}\right) + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_q^{(0)}\right)\right) \\
& -k_rk_r\nu_0\hat{\beta}^q_{s(1)}\acute{U}_s^{(0)} + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^q_{s(1)}\acute{U}_s^{(0)}\right)\right) \\
& -k_rk_s\nu_0\hat{\alpha}^s_{q(1)}\acute{U}_r^{(0)} + \mathsf{i}k_r\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{q(1)}\acute{U}_r^{(0)}\right) \\
& -k_rk_q\nu_0\hat{\beta}^r_{s(1)}\acute{U}_s^{(0)} + \mathsf{i}k_q\frac{\partial}{\partial\acute{x}_3}\left(\nu_0\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)
\end{aligned} \tag{B.154}$$

where $q, r, s = 1, 2$ and

$$\begin{aligned}
\hat{S}_3 \;=\;& -\mathsf{i}k_r\acute{U}_r^{(0)}\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)} \\
& +\mathsf{i}k_r\hat{\alpha}^r_{s(1)}\nu_0\frac{\partial\acute{U}_s^{(0)}}{\partial\acute{x}_3} + \frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{s(1)}\nu_0\frac{\partial\acute{U}_s^{(0)}}{\partial\acute{x}_3}\right) \\
& -k_rk_r\nu_0\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)} + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)\right) \\
& -k_sk_r\nu_0\hat{\alpha}^r_{3(1)}\acute{U}_s^{(0)} + \mathsf{i}k_s\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\alpha}^3_{3(1)}\acute{U}_s^{(0)}\right) \\
& +\mathsf{i}k_r\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^r_{s(1)}\acute{U}_s^{(0)}\right) + \frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial}{\partial\acute{x}_3}\left(\hat{\beta}^3_{s(1)}\acute{U}_s^{(0)}\right)\right)
\end{aligned} \tag{B.155}$$

where $r, s = 1, 2$

Eliminating the pressure and horizontal velocity components

We wish to eliminate $\hat{p}_1$ as well as $\hat{U}_1^{(1)}$ and $\hat{U}_2^{(1)}$ from the first-order equations, so that only an equation for $\hat{U}_3^{(1)} = \mathcal{W}$ remains, to be solved along with the equations

for $\mathcal{E}$ and $\mathcal{D}$.

Restating the first-order momentum equations, we have, in the case $i = 1, 2$:

$$\begin{aligned}
&\hat{U}_3^{(1)} \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3} + \mathsf{i}\left(k_1 \acute{U}_1^{(0)} + k_2 \acute{U}_2^{(0)}\right) \hat{U}_i^{(1)} \\
&+ \left(k_1 k_1 + k_2 k_2\right) \nu_0 \hat{U}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{U}_i^{(1)}}{\partial \acute{x}_3}\right) \\
&- \mathsf{i}k_i \frac{\partial \nu_0}{\partial \acute{x}_3} \hat{U}_3^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1 \frac{\partial \acute{U}_i^{(0)}}{\partial \acute{x}_3}\right) + \frac{1}{\rho} \mathsf{i}k_i \hat{p}_1 \quad = \quad \hat{S}_i
\end{aligned} \tag{B.156}$$

and in the case $i = 3$:

$$\begin{aligned}
&\mathsf{i}\left(k_1 \acute{U}_1^{(0)} + k_2 \acute{U}_2^{(0)}\right) \hat{U}_3^{(1)} + \left(k_1 k_1 + k_2 k_2\right) \nu_0 \hat{U}_3^{(1)} \\
&- \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{U}_3^{(1)}}{\partial \acute{x}_3}\right) - \frac{\partial \nu_0}{\partial \acute{x}_3} \frac{\partial \hat{U}_3^{(1)}}{\partial \acute{x}_3} \\
&- \mathsf{i}\hat{\nu}_1 \left(k_1 \frac{\partial \acute{U}_1^{(0)}}{\partial \acute{x}_3} + k_2 \frac{\partial \acute{U}_2^{(0)}}{\partial \acute{x}_3}\right) + \frac{1}{\rho} \frac{\partial \hat{p}_1}{\partial \acute{x}_3} \quad = \quad \hat{S}_3
\end{aligned} \tag{B.157}$$

We multiply Equation (B.156) by $\mathsf{i}k_i$,

$$\begin{aligned}
&\hat{U}_3^{(1)} \frac{\partial (\mathsf{i}k_i \acute{U}_i^{(0)})}{\partial \acute{x}_3} + \mathsf{i}\left(k_1 \acute{U}_1^{(0)} + k_2 \acute{U}_2^{(0)}\right) \mathsf{i}k_i \hat{U}_i^{(1)} \\
&+ \left(k_1 k_1 + k_2 k_2\right) \nu_0 \mathsf{i}k_i \hat{U}_i^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial (\mathsf{i}k_i \hat{U}_i^{(1)})}{\partial \acute{x}_3}\right) \\
&+ k_i k_i \frac{\partial \nu_0}{\partial \acute{x}_3} \hat{U}_3^{(1)} - \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1 \frac{\partial (\mathsf{i}k_i \acute{U}_i^{(0)})}{\partial \acute{x}_3}\right) - k_i k_i \frac{1}{\rho} \hat{p}_1 \quad = \quad \mathsf{i}k_i \hat{S}_i
\end{aligned} \tag{B.158}$$

The first-order continuity equation is $\mathsf{i}k_q \hat{U}_q^{(1)} = -\mathcal{W}'$ where $q = 1, 2$. We substitute this result where $\mathsf{i}k_i \hat{U}_i^{(1)}$ appears in Equation (B.158), and get

$$\begin{aligned}
&\mathcal{W} \frac{\partial A}{\partial \acute{z}} - A\mathcal{W}' - k_j k_j \nu_0 \mathcal{W}' + \frac{\partial}{\partial \acute{z}}\left(\nu_0 \mathcal{W}''\right) \\
&+ k_j k_j \frac{\partial \nu_0}{\partial \acute{z}} \mathcal{W} - \frac{\partial}{\partial \acute{z}}\left(\hat{\nu}_1 \frac{\partial A}{\partial \acute{z}}\right) - k_j k_j \frac{1}{\rho} \hat{p}_1 \quad = \quad \mathsf{i}k_j \hat{S}_j
\end{aligned} \tag{B.159}$$

where $j = 1, 2$ and where we switched to the shorthand notation.

We have thus eliminated $\hat{U}_1^{(1)}$ and $\hat{U}_2^{(1)}$. We now proceed to eliminate $\hat{p}_1$. We take the vertical derivative of Equation (B.159):

$$\begin{aligned}
&\frac{\partial}{\partial \acute{z}}\left(\mathcal{W} \frac{\partial A}{\partial \acute{z}}\right) - \frac{\partial}{\partial \acute{z}}\left(A\mathcal{W}'\right) - k_j k_j \frac{\partial}{\partial \acute{z}}\left(\nu_0 \mathcal{W}'\right) + \frac{\partial^2}{\partial \acute{z}^2}\left(\nu_0 \mathcal{W}''\right) \\
&+ k_j k_j \frac{\partial}{\partial \acute{z}}\left(\frac{\partial \nu_0}{\partial \acute{z}} \mathcal{W}\right) - \frac{\partial^2}{\partial \acute{z}^2}\left(\hat{\nu}_1 \frac{\partial A}{\partial \acute{z}}\right) - k_j k_j \frac{1}{\rho} \frac{\partial p_1}{\partial \acute{z}} = \mathsf{i}k_j \frac{\partial \hat{S}_j}{\partial \acute{z}}
\end{aligned} \tag{B.160}$$

We write the vertical momentum equation (B.157) in shorthand form:

$$A\mathcal{W} + k_q k_q \nu_0 \mathcal{W} - \frac{\partial}{\partial \acute{z}}\left(\nu_0 \mathcal{W}'\right) - \frac{\partial \nu_0}{\partial \acute{z}} \mathcal{W}' - \hat{\nu}_1 \frac{\partial A}{\partial \acute{z}} + \frac{1}{\rho} \frac{\partial \hat{p}_1}{\partial \acute{z}} = \hat{S}_3 \tag{B.161}$$

We multiply it by $k_j k_j$:

$$\begin{aligned}
&k_j k_j A\mathcal{W} + k_j k_j k_q k_q \nu_0 \mathcal{W} - k_j k_j \frac{\partial}{\partial \acute{z}}\left(\nu_0 \mathcal{W}'\right) \\
&- k_j k_j \frac{\partial \nu_0}{\partial \acute{z}} \mathcal{W}' - k_j k_j \hat{\nu}_1 \frac{\partial A}{\partial \acute{z}} + k_j k_j \frac{1}{\rho} \frac{\partial \hat{p}_1}{\partial \acute{z}} \quad = \quad k_j k_j \hat{S}_3
\end{aligned} \tag{B.162}$$

and add it to Equation (B.160):

$$\begin{aligned}
&\frac{\partial}{\partial \acute{z}}\left(\mathcal{W}\frac{\partial A}{\partial \acute{z}}\right)-\frac{\partial}{\partial \acute{z}}\left(A\mathcal{W}'\right)-k_jk_j\frac{\partial}{\partial \acute{z}}\left(\nu_0\mathcal{W}'\right)\\
+&\frac{\partial^2}{\partial \acute{z}^2}\left(\nu_0\mathcal{W}''\right)+k_jk_j\frac{\partial}{\partial \acute{z}}\left(\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}\right)-\frac{\partial^2}{\partial \acute{z}^2}\left(\hat{\nu}_1\frac{\partial A}{\partial \acute{z}}\right)\\
&+k_jk_jA\mathcal{W}+k_jk_jk_qk_q\nu_0\mathcal{W}-k_jk_j\frac{\partial}{\partial \acute{z}}\left(\nu_0\mathcal{W}'\right)\\
&-k_jk_j\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}'-k_jk_j\hat{\nu}_1\frac{\partial A}{\partial \acute{z}} \quad = \quad \mathsf{i}k_j\frac{\partial \hat{S}_j}{\partial \acute{z}}+k_jk_j\hat{S}_3
\end{aligned} \tag{B.163}$$

We now expand the derivatives of products containing $\mathcal{W}$; certain things cancel out.

$$\begin{aligned}
&\mathcal{W}\frac{\partial^2 A}{\partial \acute{z}^2}-A\mathcal{W}''-2\,k_jk_j\nu_0\mathcal{W}''\\
&-2\,k_jk_j\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}'+\mathcal{W}''\frac{\partial^2\nu_0}{\partial \acute{z}^2}+2\,\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}'''\\
&+\nu_0\mathcal{W}''''+k_jk_j\frac{\partial^2\nu_0}{\partial \acute{z}^2}\mathcal{W}-\frac{\partial^2}{\partial \acute{z}^2}\left(\hat{\nu}_1\frac{\partial A}{\partial \acute{z}}\right)\\
&+k_jk_jA\mathcal{W}+k_jk_jk_qk_q\nu_0\mathcal{W}-k_jk_j\hat{\nu}_1\frac{\partial A}{\partial \acute{z}} \quad = \quad \mathsf{i}k_j\frac{\partial \hat{S}_j}{\partial \acute{z}}+k_jk_j\hat{S}_3
\end{aligned} \tag{B.164}$$

We note that

$$\begin{aligned}
-\frac{\partial^2}{\partial \acute{z}^2}\left(\hat{\nu}_1\frac{\partial A}{\partial \acute{z}}\right) \quad &= \quad -\frac{\partial}{\partial \acute{z}}\left(\frac{\partial}{\partial \acute{z}}\left(\nu_0 2\,\frac{\hat{E}_1}{E_0}\frac{\partial A}{\partial \acute{z}}\right)\right)\\
&\quad +\frac{\partial}{\partial \acute{z}}\left(\frac{\partial}{\partial \acute{z}}\left(\nu_0\frac{\hat{\varepsilon}_1}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right)\right)\\
&= \quad -2\,\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right)\hat{E}_1-4\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right)\frac{\partial \hat{E}_1}{\partial \acute{z}}-2\,\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\frac{\partial^2\hat{E}_1}{\partial \acute{z}^2}\\
&\quad +\frac{\partial}{\partial \acute{z}}\left(\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right)\right)\hat{\varepsilon}_1+2\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right)\frac{\partial \hat{\varepsilon}_1}{\partial \acute{z}}+\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\frac{\partial^2\hat{\varepsilon}_1}{\partial \acute{z}^2}
\end{aligned}$$

and that

$$\begin{aligned}
-k_jk_j\hat{\nu}_1\frac{\partial A}{\partial \acute{z}} \quad &= \quad -k_jk_j\nu_0\left(2\,\frac{\hat{E}_1}{E_0}-\frac{\hat{\varepsilon}_1}{\varepsilon_0}\right)\frac{\partial A}{\partial \acute{z}}\\
&= \quad -2\,k_jk_j\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\mathcal{E}+k_jk_j\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\mathcal{D}
\end{aligned}$$

We finally group terms according to the order of the derivative of $\mathcal{W}$:

$$
\begin{aligned}
\frac{\partial \mathcal{W}'''}{\partial \acute{z}} = -\frac{1}{\nu_0}\Bigg[&+ 2\,\frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}''' \\
&+ \left(-A - 2\,k_j k_j \nu_0 + \frac{\partial^2 \nu_0}{\partial \acute{z}^2}\right)\mathcal{W}'' \\
&- 2\,k_j k_j \frac{\partial \nu_0}{\partial \acute{z}}\mathcal{W}' \\
&+ \left(\frac{\partial^2 A}{\partial \acute{z}^2} + k_j k_j A + k_j k_j k_q k_q \nu_0 + k_j k_j \frac{\partial^2 \nu_0}{\partial \acute{z}^2}\right)\mathcal{W} \\
&- 2\,\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\mathcal{E}'' \\
&- 4\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right)\mathcal{E}' \\
&+ \left[-2\,\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right) - 2\,k_j k_j \frac{\nu_0}{E_0}\frac{\partial A}{\partial \acute{z}}\right]\mathcal{E} \\
&+ \frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\mathcal{D}'' \\
&+ 2\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right)\mathcal{D}' \\
&+ \left[\frac{\partial^2}{\partial \acute{z}^2}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right) + k_j k_j \frac{\nu_0}{\varepsilon_0}\frac{\partial A}{\partial \acute{z}}\right]\mathcal{D} \\
&- \mathrm{i}k_j \frac{\partial \hat{S}_j}{\partial \acute{z}} \\
&- k_j k_j \hat{S}_3\Bigg]
\end{aligned}
\tag{B.165}
$$

But $\mathcal{E}'' = \frac{\partial \mathcal{E}'}{\partial \acute{z}}$ and $\mathcal{D}'' = \frac{\partial \mathcal{D}'}{\partial \acute{z}}$, and Equations (84) and (87) precisely provide expressions for these. We therefore substitute them for $\mathcal{E}''$ and $\mathcal{D}''$ in the code.

Now we spell out the source term

$$
\begin{aligned}
\mathrm{i}k_q\frac{\partial\hat{S}_q}{\partial\acute{z}}+k_qk_q\hat{S}_3 &= \frac{\partial}{\partial\acute{z}}\Big[\hat{J}_1A^2-\mathrm{i}k_q\hat{\beta}^q_{r(1)}\acute{U}^{(0)}_rA\\
&\quad+\mathrm{i}k_r\hat{\alpha}^r_{3(1)}\nu_0\frac{\partial A}{\partial\acute{z}}+\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{3(1)}\nu_0\frac{\partial A}{\partial\acute{z}}\right)\\
&\quad-k_sk_r\nu_0\hat{\alpha}^r_{s(1)}A+\mathrm{i}k_s\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{s(1)}A\right)\\
&\quad+\mathrm{i}k_r\frac{\partial}{\partial\acute{z}}\left(\nu_0\hat{\alpha}^r_{3(1)}A\right)+\frac{\partial}{\partial\acute{z}}\left(\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{3(1)}A\right)\right)\\
&\quad-\mathrm{i}k_qk_rk_r\nu_0\hat{\beta}^q_{s(1)}\acute{U}^{(0)}_s+\mathrm{i}k_q\frac{\partial}{\partial\acute{z}}\left(\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\beta}^q_{s(1)}\acute{U}^{(0)}_s\right)\right)\\
&\quad-k_qk_s\nu_0\hat{\alpha}^s_{q(1)}A+\mathrm{i}k_q\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{q(1)}A\right)\Big]\\
&\quad-\mathrm{i}k_rk_qk_q\hat{\beta}^r_{s(1)}\acute{U}^{(0)}_s\frac{\partial\nu_0}{\partial\acute{z}}-k_qk_q\frac{\partial}{\partial\acute{z}}\left(\hat{\beta}^3_{s(1)}\acute{U}^{(0)}_s\frac{\partial\nu_0}{\partial\acute{z}}\right)\\
&\quad-k_qk_qA\hat{\beta}^3_{s(1)}\acute{U}^{(0)}_s\\
&\quad+\mathrm{i}k_qk_qk_r\hat{\alpha}^r_{s(1)}\nu_0\frac{\partial\acute{U}^{(0)}_s}{\partial\acute{z}}+k_qk_q\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{s(1)}\nu_0\frac{\partial\acute{U}^{(0)}_s}{\partial\acute{z}}\right)\\
&\quad-k_qk_qk_rk_r\nu_0\hat{\beta}^3_{s(1)}\acute{U}^{(0)}_s+k_qk_q\frac{\partial}{\partial\acute{z}}\left(\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\beta}^3_{s(1)}\acute{U}^{(0)}_s\right)\right)\\
&\quad+\mathrm{i}k_qk_qk_r\nu_0\hat{\alpha}^r_{3(1)}A+k_qk_q\nu_0\frac{\partial}{\partial\acute{z}}\left(\hat{\alpha}^3_{3(1)}A\right) \qquad \text{(B.166)}
\end{aligned}
$$

B.5 Obtaining an equation for first-order vorticity

As we see in Sections B.2 and B.3, the expression $\mathrm{i}\breve{k}_q\hat{\acute{U}}^{(1)}_q$ appears in the production term of the first-order TKE and dissipation equations. This expression is simply the Fourier-transformed first-order vertical vorticity, $\hat{\acute{\omega}}^{(1)}_3$:

$$
\begin{aligned}
\mathrm{i}\breve{k}_q\hat{\acute{U}}^{(1)}_q &= \mathrm{i}\breve{k}_1\hat{\acute{U}}^{(1)}_1+\mathrm{i}\breve{k}_1\hat{\acute{U}}^{(1)}_1\\
&= -\mathrm{i}k_2\hat{\acute{U}}^{(1)}_1+\mathrm{i}\breve{k}_2\hat{\acute{U}}^{(1)}_1\\
&= \mathcal{F}\left\{-\frac{\partial\hat{\acute{U}}^{(1)}_1}{\partial\acute{x}_2}+\frac{\partial\hat{\acute{U}}^{(1)}_1}{\partial\acute{x}_1}\right\}\\
&= \mathcal{F}\left\{\acute{\omega}^{(1)}_3\right\}\\
&= \hat{\acute{\omega}}^{(1)}_3\equiv\Omega \qquad \text{(B.167)}
\end{aligned}
$$

The vorticity cannot be expressed in terms of the dependent variables $\mathcal{W},\mathcal{W}',\mathcal{W}''\ldots$ (or $\mathcal{E},\mathcal{D}$ and their derivatives, with which it has nothing to do in any direct sense). We must therefore derive an equation for $\hat{\acute{\omega}}^{(1)}_3$. We do this by multiplying the $\hat{\acute{U}}^{(1)}_q$ equation by $\mathrm{i}\breve{k}_q$ $(q=1,2)$.

We multiply the first-order horizontal momentum equation (B.156) by $\mathrm{i}\breve{k}_q$ $(q=1,2)$, and get

$$
\begin{aligned}
&\hat{\acute{U}}^{(1)}_3\frac{\partial(\mathrm{i}\breve{k}_q\acute{U}^{(0)}_q)}{\partial\acute{x}_3}+\mathrm{i}k_r\acute{U}^{(0)}_r\left(\mathrm{i}\breve{k}_q\hat{\acute{U}}^{(1)}_q\right)+k_rk_r\nu_0\left(\mathrm{i}\breve{k}_q\hat{\acute{U}}^{(1)}_q\right)\\
&-\frac{\partial}{\partial\acute{x}_3}\left(\nu_0\frac{\partial(\mathrm{i}\breve{k}_q\hat{\acute{U}}^{(1)}_q)}{\partial\acute{x}_3}\right)-\frac{\partial}{\partial\acute{x}_3}\left(\hat{\nu}_1\frac{\partial(\mathrm{i}\breve{k}_q\acute{U}^{(0)}_q)}{\partial\acute{x}_3}\right)=\mathrm{i}\breve{k}_q\hat{S}_q \qquad \text{(B.168)}
\end{aligned}
$$

into which we substitute $\hat{\omega}_3^{(1)} = \mathrm{i}\breve{k}_q \hat{U}_q^{(1)}$ and $B = \mathrm{i}\breve{k}_q \acute{U}_q^{(0)}$:

$$\begin{aligned} & \hat{U}_3^{(1)} \frac{\partial B}{\partial \acute{x}_3} + B\hat{\omega}_3^{(1)} + k_r k_r \nu_0 \hat{\omega}_3^{(1)} \\ -\frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial \hat{\omega}_3^{(1)}}{\partial \acute{x}_3}\right) - \frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1 \frac{\partial B}{\partial \acute{x}_3}\right) \quad &= \quad \mathrm{i}\breve{k}_q \hat{S}_q \end{aligned} \tag{B.169}$$

Using the shorthand notation and noting that

$$-\frac{\partial}{\partial \acute{x}_3}\left(\hat{\nu}_1 \frac{\partial B}{\partial \acute{x}_3}\right) \quad = \quad -\frac{\partial}{\partial \acute{x}_3}\left(2\,\nu_0 \frac{\hat{E}_1}{E_0}\frac{\partial B}{\partial \acute{x}_3}\right) - \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\hat{\varepsilon}_1}{\varepsilon_0}\frac{\partial B}{\partial \acute{x}_3}\right)$$

this can be written as

$$\begin{aligned} & \frac{\partial B}{\partial \acute{x}_3}\mathcal{W} + B\Omega + k_r k_r \nu_0 \Omega - \nu_0 \Omega'' - \frac{\partial \nu_0}{\partial \acute{z}}\Omega' \\ & -2\,\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{x}_3}\right)\mathcal{E} - 2\,\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{x}_3}\mathcal{E}' \\ & -\frac{\partial}{\partial \acute{x}_3}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{x}_3}\right)\mathcal{D} - \frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{x}_3}\mathcal{D}' \quad = \quad \mathrm{i}\breve{k}_q \hat{S}_q \end{aligned}$$

i.e.

$$\begin{aligned} \frac{\partial \Omega'}{\partial \acute{z}} = \frac{1}{\nu_0}\Bigg[& -\frac{\partial \nu_0}{\partial \acute{z}}\Omega' + (B + k_r k_r \nu_0)\,\Omega \\ & + \frac{\partial B}{\partial \acute{z}}\mathcal{W} \\ & -2\,\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{z}}\mathcal{E}' - 2\,\frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{E_0}\frac{\partial B}{\partial \acute{z}}\right)\mathcal{E} \\ & -\frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{z}}\mathcal{D}' - \frac{\partial}{\partial \acute{z}}\left(\frac{\nu_0}{\varepsilon_0}\frac{\partial B}{\partial \acute{z}}\right)\mathcal{D} \\ & -\mathrm{i}\breve{k}_q \hat{S}_q\Bigg] \end{aligned} \tag{B.170}$$

Carrying out the multiplication of the source term by $-\mathrm{i}\breve{k}_q$:

$$\begin{aligned} -\mathrm{i}\breve{k}_q \hat{S}_q \quad = \quad & \breve{k}_q k_r \acute{U}_r^{(0)}\left(\hat{J}_1 \acute{U}_q^{(0)} - \hat{\beta}_{r(1)}^q \acute{U}_r^{(0)}\right) \\ & + \breve{k}_q k_r \hat{\alpha}_{3(1)}^r \nu_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3} - \mathrm{i}\breve{k}_q \frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}_{3(1)}^3 \nu_0 \frac{\partial \acute{U}_q^{(0)}}{\partial \acute{x}_3}\right) \\ & + \mathrm{i}\breve{k}_q k_s k_r \nu_0 \hat{\alpha}_{s(1)}^r \acute{U}_q^{(0)} + \breve{k}_q k_s \nu_0 \frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}_{s(1)}^3 \acute{U}_q^{(0)}\right) \\ & + \breve{k}_q k_r \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \hat{\alpha}_{3(1)}^r \acute{U}_q^{(0)}\right) - \mathrm{i}\breve{k}_q \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}_{3(1)}^3 \acute{U}_q^{(0)}\right)\right) \\ & + \mathrm{i}\breve{k}_q k_r k_r \nu_0 \hat{\beta}_{p(1)}^q \acute{U}_p^{(0)} - \mathrm{i}\breve{k}_q \frac{\partial}{\partial \acute{x}_3}\left(\nu_0 \frac{\partial}{\partial \acute{x}_3}\left(\hat{\beta}_{p(1)}^q \acute{U}_p^{(0)}\right)\right) \\ & + \mathrm{i}\breve{k}_q k_r k_s \nu_0 \hat{\alpha}_{q(1)}^s \acute{U}_r^{(0)} + \breve{k}_q k_r \nu_0 \frac{\partial}{\partial \acute{x}_3}\left(\hat{\alpha}_{q(1)}^3 \acute{U}_r^{(0)}\right) \end{aligned} \tag{B.171}$$

where $q, r, s = 1, 2$

C Analytical solution

This analytical solution to the first-order ODEs (Equation 102) was fortuitously discovered in the course of performing manipulations using Mathematica.

$$
\begin{aligned}
\mathcal{W} &= \mathrm{i}U_{*0}k(1+k\acute{z})\ln\left(\frac{z_0+\acute{z}}{z_0}\right)e^{-k\acute{z}}\kappa^{-1} \\
\mathcal{W}' &= -\frac{\mathrm{i}U_{*0}k(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})} - \frac{\mathrm{i}U_{*0}k^2\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} + \frac{\mathrm{i}U_{*0}k^2(1+k\acute{z})\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} \\
\mathcal{W}'' &= -\frac{2\mathrm{i}U_{*0}k^2}{e^{k\acute{z}}\kappa(z_0+\acute{z})} + \frac{\mathrm{i}U_{*0}k(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2} + \frac{2\mathrm{i}U_{*0}k^2(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})} + \frac{2\mathrm{i}U_{*0}k^3\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} \\
&\quad - \frac{\mathrm{i}U_{*0}k^3(1+k\acute{z})\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} \\
\mathcal{W}''' &= \frac{3\mathrm{i}U_{*0}k^2}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2} + \frac{6\mathrm{i}U_{*0}k^3}{e^{k\acute{z}}\kappa(z_0+\acute{z})} - \frac{2\mathrm{i}U_{*0}k(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^3} - \frac{3\mathrm{i}U_{*0}k^2(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2} \\
&\quad - \frac{3\mathrm{i}U_{*0}k^3(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})} - \frac{3\mathrm{i}U_{*0}k^4\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} + \frac{\mathrm{i}U_{*0}k^4(1+k\acute{z})\ln\left(\frac{z_0+\acute{z}}{z_0}\right)}{e^{k\acute{z}}\kappa} \\
\mathcal{E} &= 0 \\
\mathcal{E}' &= 0 \\
\mathcal{D} &= -\frac{U_{*0}^3(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2} \\
\mathcal{D}' &= -\frac{U_{*0}^3k}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2} + \frac{2U_{*0}^3(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^3} + \frac{U_{*0}^3k(1+k\acute{z})}{e^{k\acute{z}}\kappa(z_0+\acute{z})^2}
\end{aligned}
$$

References

Allen, T. and Brown, A. R.: 1971, Analysis of a laser doppler anemometer, *Journal of Physics E: Scientific Instruments* **4**(7), 505–511.

Allen, T. and Brown, A. R.: 2002, Large-eddy simulation of turbulent separated flow over rough hills, *Boundary-Layer Meteorol.* **102**(2), 117–198.

Apsley, D. D. and Castro, I. P.: 1997, A limited-length-scale k-ε model for the neutral and stably-stratified atmospheric boundary layer, *Boundary-Layer Meteorol.* **83**(1), 75–98.

Astrup, P., Mikkelsen, T. and Jensen, N. O.: 1997, A fast model for mean and turbulent wind characteristics over terrain with mixed surface roughness, *Radiat. Prot. Dosim.* **73**(1–4), 257–260.

Athanassiadou, M. and Castro, I. P.: 2001, Neutral flow over a series of rough hills: a laboratory experiment, *Boundary-Layer Meteorol.* **101**(1), 1–30.

Ayotte, K. W. and Taylor, P. A.: 1995, A mixed spectral finite-difference model of neutral planetary boundary-layer flow over topography, *J. Atmos. Sci.* **52**(20), 3523–3537.

Ayotte, K. W., Xu, D. and Taylor, P. A.: 1994, The impact of turbulence closure schemes on predictions of the mixed spectral finite-difference model for flow over topography, *Boundary-Layer Meteorol.* **68**(1–2), 1–33.

Belcher, S. E. and Hunt, J. C. R.: 1998, Turbulent flow over hills and waves, *Annu. Rev. Fluid Mech.* **30**, 507–538.

Beljaars, A. C. M., Walmsley, J. L. and Taylor, P. A.: 1987, A mixed spectral finite-difference model for neutrally stratified boundary-layer flow over roughness changes and topography, *Boundary-Layer Meteorol.* **38**(3), 273–303.

Benoit, R., Desgagne, M., Pellerin, P., Pellerin, S., Chartier, Y. and Desjardins, S.: 1997, The canadian mc2: A semi-lagrangian, semi-implicit wideband atmospheric model suited for finescale process studies and simulation, *Mon. Weather Rev.* **125**(10), 2382–2415.

Betz, A.: 1920, Das Maximum der theoretisch möglichen Ausnutzung des Windes durch Windmotoren, *Zeitschrift für das gesamte Turbinenwesen* **26**, 307–309.

Bradbury, L. J. S. and Castro, I. P.: 1971, A pulsed-wire technique for velocity measurements in highly turbulent flows, *J. Fluid Mech.* **49**, 657–691.

Brown, A. R., Hobson, J. M. and Wood, N.: 2001, Large-eddy simulation of neutral turbulent flow over rough sinusoidal ridges, *Boundary-Layer Meteorol.* **98**(3), 411–441.

Brown, A. R. and Wood, N.: 2001, Turbulent form drag on anisotropic three-dimensional orography, *Boundary-Layer Meteorol.* **101**(2), 229–241.

Businger, J. A.: 1973, Turbulent transfer in the atmospheric surface layer, *in* D. A. Haugen (ed.), *Workshop on Micrometeorology*, American Meteorological Society, Washington DC, chapter 2.

Castelli, T. S., Ferrero, E. and Anfossi, D.: 2001, Turbulence closures in neutral boundary layers over complex terrain, *Boundary-Layer Meteorol.* **100**(3), 405–419.

Castro, F. A., Palma, J. M. L. M. and Silva Lopez, A.: 2003, Simulation of the Askervein flow. part 1: Reynolds averaged Navier-Stokes equations ($k-\epsilon$ turbulence model), *Boundary-Layer Meteorol.* **107**(3), 501–530.

Compaq Computer Corporation: 2000, Compaq Visual Fortran Standard Edition 6.5.0.

Detering, H. W. and Etling, D.: 1985, Application of the $E-\varepsilon$ turbulence model to the atmospheric boundary layer, *Boundary-Layer Meteorol.* **33**(2), 113–133.

Ding, L. and Street, R. L.: 2003, Numerical study of the wake structure behind a three-dimensional hill, *J. Atmos. Sci.* **60**(14), 1678–1690.

Eidsvik, K. J.: 2005, A system for wind power estimation in mountainous terrain. prediction of Askervein hill data, *Wind Energy* **8**(2), 237–249.

Emeis, S., Courtney, M. S., Højstrup, J. and Jensen, N. O.: 1993, Hjardemål experiment data report, *Technical Report Risø–M–2289(EN)*, Risø National Laboratory, Roskilde, Denmark.

Errico, R. and Baumhefner, D.: 1987, Predictability experiments using a high-resolution limited-area model, *Mon. Weather Rev.* **115**(2), 488–504.

Ferziger, J. H. and Perić, M.: 2002, *Computational Methods for Fluid Dynamics*, 3rd edn, Springer Verlag, Berlin.

Foudhil, H., Brunet, Y. and Caltagirone, J.: 2005, A fine-scale $k-\epsilon$ model for atmospheric flow over heterogeneous landscapes, *Environ. Fluid Mech.* **5**(3), 247–265.

Frigo, M. and Johnson, S. G.: 1998, FFTW: An adaptive software architecture for the FFT, *Proceedings of the 1998 ICASSP conference*, Vol. 3, pp. 1381–1384.

Frigo, M. and Johnson, S. G.: 2005, The design and implementation of FFTW3, *Proceedings of the IEEE*, Vol. 93, pp. 216–231.

Frost, W., Maus, J. R. and Fichtl, G. H.: 1974, A boundary-layer analysis of atmospheric motion over a semi-elliptical surface obstruction, *Boundary-Layer Meteorol.* **7**(2), 165–184.

Global Wind Energy Council: 2005, Global wind 2005 report. Available at `http://www.gwec.net/fileadmin/documents/Publications/Global\_WindPower\_05\_Report.pdf`.

Gong, W., Taylor, P. A. and Dörnback, A.: 1996, Turbulent boundary-layer flow over fixed aerodynamically rough two-dimensional sinusoidal waves, *J. Fluid Mech.* **312**, 1–37.

Hancock, P. E.: 2004, Pulsed-wire measurements in the near-wall layer in a reattaching separated flow, *Exp. Fluids* **37**(3), 323–330.

Hunt, J. C. R., Leibovich, S. and Richards, K. J.: 1988, Turbulent shear flows over low hills, *Q. J. R. Meteorol. Soc.* **114**(484), 1435–1470.

Intel Corporation: 2005, Intel Visual Fortran Compiler 9.1. `http://www.intel.com/cd/software/products/asmo-na/eng/compilers/278834.htm`.

Jackson, P. S. and Hunt, J. C. R.: 1975, Turbulent wind flow over a low hill, *Q. J. R. Meteorol. Soc.* **101**(430), 929–955.

Jensen, N. O.: 1983, Escarpment induced flow perturbations, a comparison of measurements and theory, *J. Wind Eng. Ind. Aerodyn.* **15**, 243–251.

Jones, W. P. and Launder, B.: 1972, The prediction of laminarization with a two-equation model of turbulence, *Int. J. Heat Mass Transf.* **15**(2), 301–314.

Jørgensen, B. H.: 2003a, The horizontally homogeneous model equations of incompressible atmospheric flow in general orthogonal coordinates, *Technical Report Risø–R–1446(EN)*, Risø National Laboratory, Roskilde, Denmark.

Jørgensen, B. H.: 2003b, Tensor formulation of the model equations on strong conservation form for an incompressible flow in general coordinates, *Technical Report Risø–R–1445(EN)*, Risø National Laboratory, Roskilde, Denmark.

Jørgensen, B. H., Ott, S., Sørensen, N. N., Mann, J. and Badger, J.: 2006, Computational methods in wind power meteorology, *Technical Report Risø–R–1560(EN)*, Risø National Laboratory, Roskilde, Denmark.

Kaimal, J. C. and Finnigan, J. J.: 1994, *Atmospheric Boundary Layer Flows*, Oxford University Press, chapter 5.

Kim, H. G. and Patel, V. C.: 2000, Test of turbulence models for wind flow over terrain with separation and recirculation, *Boundary-Layer Meteorol.* **94**(1), 5–21.

Kim, H. G., Patel, V. C. and Lee, C. M.: 2000, Numerical simulation of wind flow over hilly terrain, *J. Wind Eng. Ind. Aerodyn.* **87**(1), 45–60.

Launder, B. E. and Sharma, B. I.: 1974, Application of the energy dissipation model of turbulence to the calculation of flow near a spinning disc, *Lett. Heat Mass Transfer* **1**(2), 131–138.

Launder, B. E. and Spalding, D. B.: 1974, The numerical computation of turbulent flows, *Comp. Meth. Appl. Mech. Engineer.* **3**(2), 269–289.

Lettau, H. H.: 1962, Theoretical wind spirals in the boundary layer of a barotropic atmosphere, *Beitr. Phys. Atmos.* **35**, 195–212.

Mason, P. J. and King, J. C.: 1985, Measurements and predictions of flow and turbulence over an isolated hill of moderate slope, *Q. J. R. Meteorol. Soc.* **111**(468), 617–640.

Mason, P. J. and Sykes, R. I.: 1979, Flow over an isolated hill of moderate slope, *Q. J. R. Meteorol. Soc.* **105**(444), 383–395.

McLaughlin, D. K. and Tiederman, W. G.: 1973, Biasing correction for individual realization of laser anemometer measurements in turbulent flows, *Phys. Fluids* **16**(12), 2082–2088.

Monin, A. S. and Yaglom, A. M.: 1971, *Statistical fluid mechanics – Mechanics of turbulence*, MIT Press, Cambridge, MA.

Myllerup, L., Corbett, J. F. and Jørgensen, B. H.: 2005, The new WAsP flow model, *Technical Report Risø–I–2282(EN)*, Risø National Laboratory, Roskilde, Denmark.

Newley, T. J.: 1985, *Turbulent airflow over hills*, PhD thesis, University of Cambridge.

Perot, J. B.: 1993, An analysis of the fractional step method, *Journal of Computational Physics* **108**(1), 51–58.

Petersen, E. L., Mortensen, N. G., Landberg, L., Højstrup, J. and Frank, H. P.: 1998a, Wind power meteorology. Part I: climate and turbulence, *Wind Energy* **1**(1), 2–22.

Petersen, E. L., Mortensen, N. G., Landberg, L., Højstrup, J. and Frank, H. P.: 1998b, Wind power meteorology. Part II: siting and models, *Wind Energy* **1**(2), 55–72.

Press, W. H., Teukolsky, S. A., Vetterling, W. T. and Flannery, B. P.: 1992, *Numerical Recipes in Fortran 77*, 2nd edn, Cambridge University Press.

Raithby, G. D., Stubley, G. D. and Taylor, P. A.: 1987, The Askervein hill project: A finite control volume prediction on three-dimensional flows over the hill, *Boundary-Layer Meteorol.* **39**(3), 15–39.

Risø National Laboratory: 1987, Wind Atlas Analysis and Application Program (WAsP). `http://www.wasp.dk/`.

Risø National Laboratory: 1999, WAsP Engineering. `http://www.waspengineering.dk/`.

Ross, A. N., Arnold, S., Vosper, S. B., Mobbs, S. D., Dixon, N. and Robins, A. G.: 2004, A comparison of wind-tunnel experiments and numerical simulations of neutral and stratified flow over a hill, *Boundary-Layer Meteorol.* **113**(3), 427–459.

Salmon, J. R., Bowen, A. J., Hoff, A. M., Johnson, R., Mickle, R. E., Taylor, P. A., Tetzlaff, G. and Walmsley, J. L.: 1988, The Askervein hill project: Mean wind variations at fixed heights above ground, *Boundary-Layer Meteorol.* **43**(3), 247–271.

Salmon, J. R., Teunissen, H. W., Mickle, R. E. and Taylor: 1988, The Kettles hill project: Field observations, wind-tunnel simulations and numerical model preductions for flow over a low hill, *Boundary-Layer Meteorol.* **43**(4), 309–343.

Schober, M., Hancock, P. E. and Siller, H.: 1998, Pulsed-wire anemometry near walls, *Exp. Fluids* **25**(2), 151–159.

Sogachev, A. and Panferov, O.: 2006, Modification of two-equation models to account for plant drag, *Boundary-Layer Meteorol.* **121**(2), 229–266.

Sørensen, N. N.: 1995, *General Purpose Flow Solver Applied to Flow over Hills*, PhD thesis, Technical University of Denmark, Kgs. Lyngby, Denmark. Also available as Technical report Risø–R–827(EN), Risø National Laboratory, Roskilde, Denmark.

Sykes, R. I.: 1980, An asymptotic theory of incompressible turbulent boundary-layer flow over a small hump, *J. Fluid Mech.* **101**, 647–670.

Taylor, P. A.: 1977, Some numerical solutions of surface boundary-layer flow above gentle topography, *Boundary-Layer Meteorol.* **11**(4), 439–465.

Taylor, P. A.: 1998, Turbulent boundary-layer flow over low and moderate slope hills, *J. Wind Eng. Ind. Aerodyn.* **74–76**, 25–47.

Taylor, P. A. and Gent, P. R.: 1974, A model of atmospheric boundary-layer flow above an isolated two dimensional hill: an example of flow over gentle topography, *Boundary-Layer Meteorol.* **7**(3), 349–362.

Taylor, P. A., Mason, P. J. and Bradley, E. F.: 1987, Boundary layer flow over hills – A review, *Boundary-Layer Meteorol.* **39**(1–2), 107–132.

Taylor, P. A. and Teunissen, H. W.: 1987, Askervein hill project: Overview and background data, *Boundary-Layer Meteorol.* **39**(1–2), 15–39.

Taylor, P. A., Walmsley, J. L. and Salmon, J. R.: 1983, A simple model of neutrally stratified boundary-layer flow over real terrain incorporating wavenumber-dependent scaling, *Boundary-Layer Meteorol.* **26**(2), 169–189.

Troen, I. and de Baas, A.: 1986, A spectral diagnostic model for wind flow simulation in complex terrain, *Proceedings of the European Wind Energy Association Conference & Exhibition*, Rome, pp. 37–41.

Troen, I. and Petersen, E. L.: 1989, *European Wind Atlas*, Risø National Laboratory, Roskilde, Denmark.

Uchida, T. and Ohya, Y.: 2003, Large-eddy simulation of turbulent airflow over complex terrain, *J. Wind Eng. Ind. Aerodyn.* **91**(1–2), 219–229.

Uchida, T. and Ohya, Y.: 2006, Application of les technique to diagnosis of wind farm by using high-resolution elevation data, *JSME Int. J. Ser. B-Fluids Therm. Eng.* **49**(3), 567–575.

Undheim, O., Andersson, H. I. and Berge, E.: 2006, Non-linear, microscale modelling of the flow over Askervein hill, *Boundary-Layer Meteorol.* **120**(3), 477–495.

Walmsley, J. L., Salmon, J. R. and Taylor, P. A.: 1982, On the application of a model of boundary-layer flow over low hills to real terrain, *Boundary-Layer Meteorol.* **23**(1), 17–46.

Walmsley, J. L. and Taylor, P. A.: 1996, Boundary-layer flow over topography: Impacts of the Askervein study, *Boundary-Layer Meteorol.* **78**(3–4), 291–320.

Weibull, W.: 1951, A statistical distribution function of wide applicability, *J. Appl. Mech.-Trans. ASME* **18**(3), 293–297.

Wilcox, D. C.: 1998, *Turbulence Modelling for CFD*, 2nd edn, DCW Industries, 5354 Palm Drive, La Cañada, CA 91011.

Wolfram Research, Inc.: 2005, Mathematica 5.2. `http://www.wolfram.com/`.

Wood, N.: 1995, The onset of separation in neutral, turbulent flow over hills, *Boundary-Layer Meteorol.* **76**(1–2), 137–164.

Wood, N.: 2000, Wind flow over complex terrain: a historical perspective and the prospect for large-eddy simulation modelling, *Boundary-Layer Meteorol.* **96**(1–2), 11–32.

Wood, N. and Mason, P. J.: 1993, The pressure force induced by neutral, turbulent flow over hills, *Q. J. R. Meteorol. Soc.* **119**(514), 1233–1267.

Xu, D., Ayotte, K. W. and Taylor, P. A.: 1994, Development of a nonlinear mixed spectral finite difference model for turbulent boundary-layer flow over topography, *Boundary-Layer Meteorol.* **70**(4), 341–367.

Xu, D. and Taylor, P. A.: 1992, A nonlinear extension of the mixed spectral finite difference model for neutrally stratified turbulent flow over topography, *Boundary-Layer Meteorol.* **59**(1–2), 177–186.

Zak, J. A.: 2003, Atmospheric boundary layer sensors for application in a wake vortex advisory system, *Technical Report NASA/CR–2003–212175*, NASA Scientific and Technical Information Program Office, Hanover, MD.

Zeman, O. and Jensen, N. O.: 1987, Modification of turbulence characteristics in flow over hills, *Q. J. R. Meteorol. Soc.* **113**(475), 55–80.

Risø's research is aimed at solving concrete problems in the society.

Research targets are set through continuous dialogue with business, the political system and researchers.

The effects of our research are sustainable energy supply and new technology for the health sector.